NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

ANALYSIS OF A DIGITAL TECHNIQUE FOR
FREQUENCY TRANSPOSITION OF SPEECH

by

Vincent DiGirolamo

September 1985

Thesis Advisor: Paul H. Moose

Approved for public release: distribution is unlimited.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Analysis of a Digital Technique for Frequency Transposition of Speech		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis September 1985
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Vincent DiGirolamo		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		12. REPORT DATE September 1985
		13. NUMBER OF PAGES 99 pages
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Frequency Transposition, Speech, Hearing Impaired, Pole-Shifting, Frequency Content, Linear Predictive Coding, Reflection Coefficient, LPC Parameters, Speech Processing

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Frequency transposition is the process of raising or lowering the frequency content (pitch) of an audio signal. The hearing impaired community has the greatest interest in the applications of frequency transposition. Though several analog and digital frequency transposing hearing aid systems have been built and tested, this thesis investigates a possible digital processing alternative. Pole shifting, in the z-domain, of an

20. (Continued)

autoregressive (all pole) model of speech was proven to be a viable theory for changing frequency content. Since linear predictive coding (LPC) techniques are used to code, analyze and synthesize speech, with the resulting LPC coefficients related to the coefficients of an equivalent autoregressive model, a linear relationship between LPC coefficients and frequency transposition is explored. This theoretical relationship is first established using a pure sine wave and then is extended into processing speech. The resulting speech synthesis experiments failed to substantiate the conjectures of this thesis. However, future research avenues are suggested that may lead toward a viable approach to transpose speech.

Analysis of a Digital Technique for Frequency Transposition of Speech

by

Vincent DiGirolamo
Lieutenant, United States Navy
B.S., United States Naval Academy, 1978

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 1985

ABSTRACT

Frequency transposition is the process of raising or lowering the frequency content (pitch) of an audio signal. The hearing impaired community has the greatest interest in the applications of frequency transposition. Though several analog and digital frequency transposing hearing aid systems have been built and tested, this thesis investigates a possible digital processing alternative. Pole shifting, in the z-domain, of an autoregressive (all pole) model of speech was proven to be a viable theory for changing frequency content. Since linear predictive coding (LPC) techniques are used to code, analyze and synthesize speech, with the resulting LPC coefficients related to the coefficients of an equivalent autoregressive model, a linear relationship between LPC coefficients and frequency transposition is explored. This theoretical relationship is first established using a pure sine wave and then is extended into processing speech. The resulting speech synthesis experiments failed to substantiate the conjectures of this thesis. However, future research avenues are suggested that may lead toward a viable approach to transpose speech.

TABLE OF CONTENTS

ACKNOWLEDGMENTS

This work is the result of the efforts and support of many people. I attribute the accomplishment of this research and every sacrifice that lead to it's completion to my loving wife, Diane. Without your support and encouragement over the past two and a half years, I could not have done it. I love you.

My thanks and appreciation goes to Professor Moose who took an interest in this topic and allowed me to pursue it my way. That means a lot to me. Also, to Professor Madan who patiently worked with me on my many schemes for data acquisition, and endured with me as my second reader until its completion, I am grateful.

To my friends, Dennis Poulos and Alan Farmer, who labored with me in my many academic trials, I cherish many fond memories and am not quick to forget, fair winds.

Finally, in the most critical moments of this research, Val Johnson and Bob Scott provided me with the means of transforming and transfering the speech data necessary for the speech processing portion of this research. Without your rescue efforts, I wouldn't be writing these words.

Though words are not always adequate, your efforts will always be appreciated, God speed, until we meet again.

I. INTRODUCTION

A. BACKGROUND

Adjusting the frequency content or pitch of a signal is a topic researched within the audio field. The hearing impaired community has the greatest interest in the applications of frequency modification or transposition techniques. This is due to their need for auditory speech-processing aids.

Auditory speech-processing aids are divided into two groups: those which involve nonradical processing of the speech signal, with the speech still intelligible to a person with normal hearing, and those which involve radical re-coding of the speech signal [Ref. 1:pp. 547-557].

An example of radical recoding involves such systems as cochlear implants where the normal speech signal is processed into a series of vibrations that the brain interprets as sound. Individuals who have this type of aid surgically inserted in their cochlear must learn a completely different language than a person with normal hearing. Examples of nonradical processing aids include the most widely used amplifier aids and the less familiar frequency lowering devices or frequency transposition systems.

Most hearing aids amplify sound. Some aids may amplify or soften certain frequencies, while others transmit sound from the aid on one ear to the aid on the other ear. Their primary purpose, in either case, is to amplify everything they are capable of sensing. In this thesis, however, we are interested in developing an algorithm that may someday drive an aid which lowers the frequency content and preserves the intelligibility of the speech signal.

B. FREQUENCY MODIFICATION

Pickett [Ref. 2:pp. 191-194] categorizes two basic methods that have been used for frequency lowering:

1. Frequency transposition, where a portion of the signal is separated out and resynthesized in a lower frequency band.

2. Frequency division, where the frequency of the signal is reduced by a fixed ratio.

All of the methods involve signal distortion. Signal distortion tends to increase with greater frequency shifts. Here we are concerned primarily with the idea of moderate frequency transposition, where the signal is shifted without major distortions in the information content.

The earliest known suggestion of frequency lowering was by Perwitschky (1925). The earliest transposing hearing aid was built and tested by Johansson (1955). Since then, there have been several other systems built and tested, but considering the advances and trends of current technology,

research in the area of frequency transposition of speech has not been productive.

Frequency transposition systems have utilized analog techniques such as frequency modulation (shifting an upper band to a lower band); frequency division (a slow playback of a tape recorded signal); and digital techniques such as sampling distortion (omitting segments of recorded speech), and doppler (the delaying of the incoming signal). Though these methods have been developed and extensively tested, the digital approach presented here may produce, all together, different results.

Pickett confirms that the possibilities for usable frequency shifting algorithms have not been explored extensively enough to make recommendations for practice [Ref. 2:p. 193]. The research needs in this area include obtaining new information on the potential for digital re-coding, exploring the principles of transposition, finding which general cues can be sent in this way, finding the optimum parameters, and examining what system can be built that meets our general and specific needs.

C. A NEW TECHNIQUE FOR FREQUENCY TRANSPOSITION

Recently, Hall [Ref. 3:p. 56] postulated that pole shifting in the z-domain using an auto-regressive (all pole) model of speech may be a possible option for frequency lowering. He used linear predictive coding (LPC) techniques

to process the speech to determine if pole shifting was a viable option. His experimental results were positive because he was able to create a change in pitch on the input speech segment.

This thesis is an extension of Hall's research. It ventures beyond the frequency domain model, and works directly with the linear predictive time domain model. It was postulated that a linear relationship exists between frequency content and the reflection coefficients determined using LPC. Once this theory has been postulated, a speech processing experiment was undertaken to determine if the conjectures made were plausible.

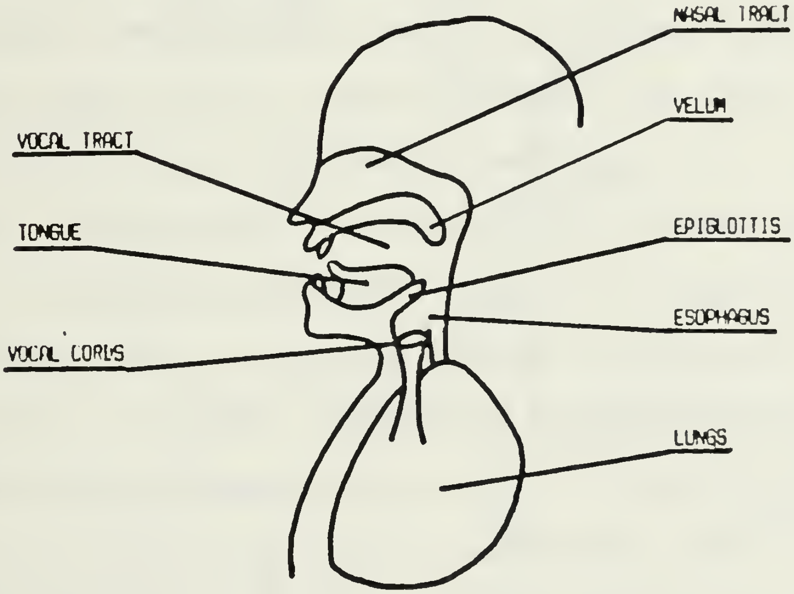
In this report linear prediction is introduced, the particular algorithms used to process the data are explained, and experimental research was carried out. Identical phrases of speech, spoken at different pitch levels by the same speaker, are sampled and processed. Possible patterns existing between the different pitch segments of speech and their linear predictive coefficients are analyzed.

The results of this research indicate that there is no linear relationship that exists between the frequency content of speech and the LPC reflection coefficients, and recommendations are made for continued analysis concerning linear predictive coding and the frequency transposition of speech.

II. MODELING SPEECH PRODUCTION

A. INTRODUCTION

In order to understand speech reproduction and synthesis, it is useful to consider some of the basic elements that combine to produce speech. The most elementary model used to explain the production of speech is the human model illustrated below as Figure 1.



Human Speech Production System [Ref. 4:p. 42].

Figure 1.

The lungs produce the air flow necessary to begin the generation of sound. The vocal cords, tongue, mouth, lips and nasal tract combine their different properties to shape the airflow to produce the speech waveform we hear.

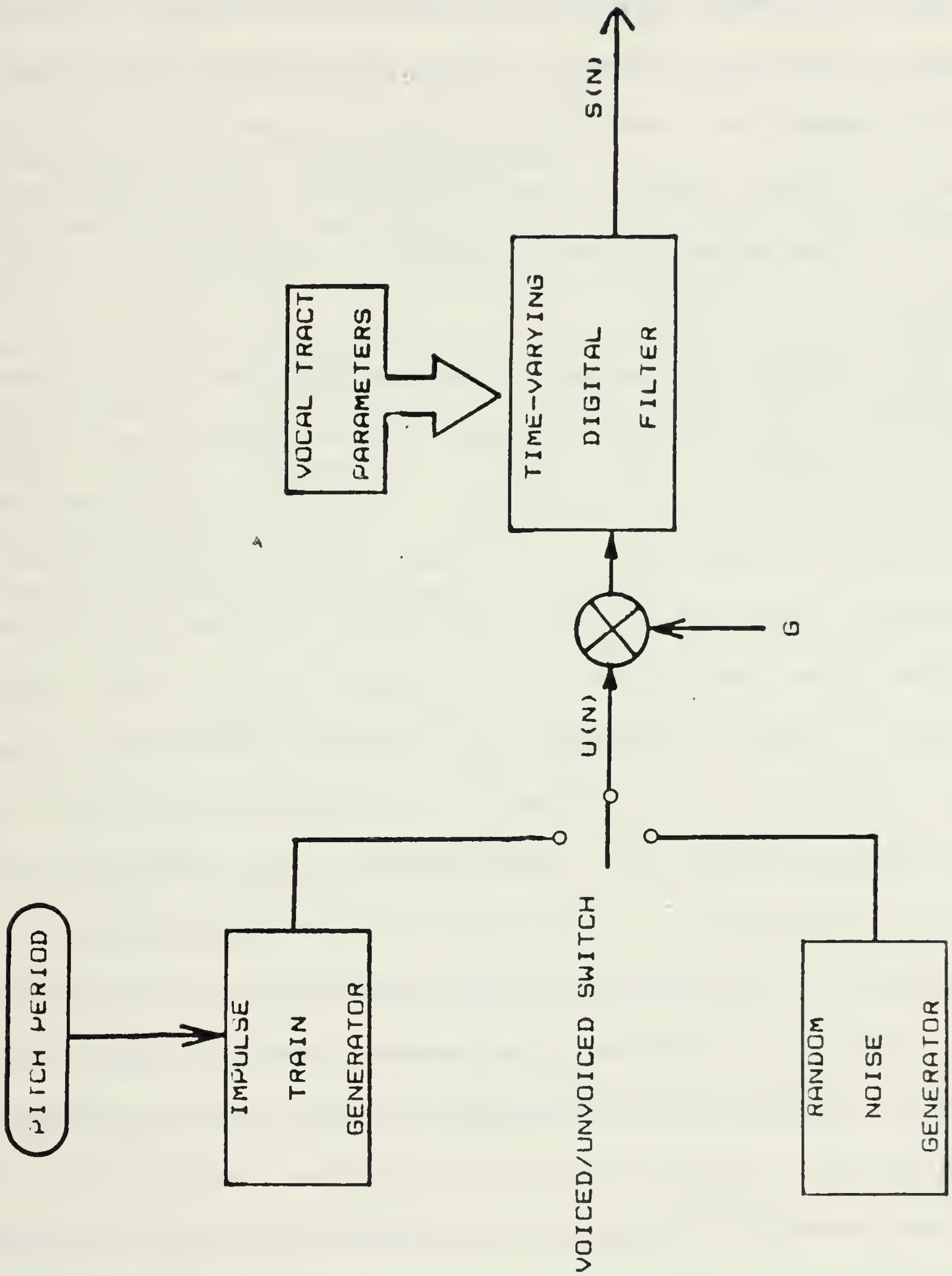
B. THE SPEECH PRODUCTION MODEL

Evans [Ref. 4:pp. 40-45] relates the several human functions to mechanical models. This is standard practice and a widely accepted approach to speech production modeling. He states that the lungs are the excitation source for the vocal and nasal tract areas. An excitation source can either be modeled as a pulse train generator or a random number generator when reproducing speech.

In the case of voiced sounds (ie. consonants, vowels or nasal sounds), the air released by the lungs is periodically modulated by vibrations from the vocal cords, glottis, and velum. Thus the excitation model in this case is a pulse generator. In the case of unvoiced sounds (ie. sh, sss, fff) which require no vibrations to be produced, the modeled excitation source is a random number generator.

Both excitation sources produce a quasi-periodic wave form that we recognize as speech. That is, the period of the wave form varies with time depending on the sound being produced. This phenomena is most obvious in the production of voiced or vibrated sounds. Figure 2, a general discrete-time model of the human speech process, illustrates this point more clearly. Here we have represented the vocal tract model as a time-varying digital filter.

Note that the pulse train has an input labeled pitch period. This input determines when the pulses will be



Discrete-time Model for Speech Production [Ref. 4:p. 43].

Figure 2.

emitted from the pulse generator and at what periodicity. This is only necessary for voiced speech.

The unvoiced speech is a continuous stream of random numbers commonly referred to as white noise. The flow of random numbers may produce a seemingly quasi-periodic sound, however, since they are usually of such short duration, we consider the sound to be continuous and constant, and not periodic.

Each speech waveform has a specific amount of energy. The energy contained within each utterance of a set duration will be referred to as gain (G). This is what gives speech its body or quality. It also aids reproduction by indicating the intensity or inflection of the voice signal.

Once the voiced or unvoiced decision is made and an energy or gain is assigned, the scaled excitation function drives the vocal tract model. In a phone interview with James Kaiser of Bell Laboratories, he mentioned that current thinking in the area of speech reproduction has refocused its attention on this portion of the model and that there is a movement to more clearly describe the physics behind the different physical contributors of speech.

This vocal tract model is driven by the excitation and energy function and controlled by time varying vocal tract parameters. These vocal tract parameters adjust the vocal tract model to yield the desired output waveform. By

replacing the vocal tract model with an equivalent time-varying digital filter that models the vocal tract model's response, we are able to step right into the next phase of synthetic speech reproduction.

C. DIGITAL FILTER REPRESENTATION

Although speech is modeled most efficiently by poles and zeros, it may also be modeled accurately by an auto-regressive (all pole) filter if the order of the filter is large enough. For example, a tenth order auto-regressive filter will accurately model most audible sounds. Therefore, the transfer function (H(z)) of the digital filter in Figure 4. is shown as Eq. 1-1.

$$H(z) = \frac{G}{1 - \sum_{k=1}^{p} a_k z^k} \qquad (2\text{-}1)$$

where p is the order of the filter, G is the gain, and a_k is the filter coefficient.

G and a_k are the time-varying vocal tract parameters for this filter. For a given segment of time (i.e., 10 milli-seconds) the vocal tract parameters are constant. However, stringing these segments together in rapid succession to produce a one second interval of speech, the parameters will change 100 times. This is why they are referred to as time varying; they vary over a short period of time.

The type of digital filter used in Figure 2 is arbitrary. It is the concept behind the diagram that counts. For the purposes of this research, the properties and attributes of a time-varying lattice filter are best because they lend themselves well to linear predictive coding implementation.

III. LINEAR PREDICTION THEORY

A. WHY LINEAR PREDICTION?

Although spectral analysis is a well-known technique for studying signals, its application to speech signals suffers from a number of serious limitations arising from the nonstationary as well as the quasiperiodic properties of the speech wave. By modeling the speech wave itself, rather than its spectrum, we avoid the problems inherent in frequency-domain methods.

For instance, traditional Fourier analysis methods require a relatively long speech segment to provide adequate spectral resolution. As a result, rapidly changing speech events cannot be accurately followed [Ref. 5:pp. 276-294].

Linear predictive coding is applicable to a wide range of research problems including speech production and perception. One of the main objectives in any speech processing technique is the synthesis of speech which is indistinguishable from normal human speech.

Atal noted that much can be learned about the information-carrying structure of speech by selectively altering the properties of the speech signal. He also stated that LPC techniques can serve as a tool for modifying the acoustic properties of the speech signal [Ref. 5:p.276]. These are exactly the intentions of this thesis: to modify

the speech signal by investigating the properties of the information carrying structure.

The remainder of this chapter is a summary of linear prediction theory. The major portion of this section is extracted from Makhoul's tutorial review on linear prediction [Ref. 6:pp. 124-143], and will be based on an intuitive approach, with emphasis on the clarity of ideas rather than mathematical rigor.

B. LPC THEORY

In applying time series analysis, each continuous signal s(t) is sampled to obtain a discrete-time signal s(nT), also known as a time series, where n is an integer variable and T is the sampling interval. The sampling frequency is then $f_s=1/T$. Note that s(nT) will be represented as s_n in this discussion.

The signal s_n is considered to be the output of some system with some unknown input u_n such that the following relation holds:

$$s_n = -\sum_{k=1}^{p} a_k s_{n-k} + G \sum_{l=0}^{q} b_l u_n \qquad (3\text{-}1)$$

where a_k, b_l, and the gain G are the parameters of the hypothesized system. This equation says that the 'output' s_n is a linear combination of past outputs and present and past inputs. That is, the signal s_n is predictable from

linear combinations of past outputs and inputs. Hence the name linear prediction.

C. PARAMETER ESTIMATION

In the all-pole model, we assume that the signal s_n is given as a linear combination of its past values and some current input u_n :

$$s_n = -\sum_{k=1}^{p} a_k s_{n-k} + G u_n \qquad (3\text{-}2)$$

which yields the following frequency domain transfer function

$$H(z) = \frac{G}{1 + \sum_{k=1}^{p} a_k z^{-k}} \qquad (3\text{-}3)$$

Given a particular signal s_n, the problem is to determine the predictor coefficients $\{a_k\}$ and the gain G in some manner.

1. Method of Least Squares

Here we assume that the input u_n is totally unknown, which is the case of speech analysis. Therefore, the signal s_n can at best be approximately predicted from a linearly weighted summation of past samples. Let the approximation of s_n be $\tilde{s}_n$, where

$$\hat{s}_n = - \sum_{k=1}^{p} a_k s_{n-k} \tag{3-4}$$

Then the error between the actual value s_n and the predicted value $\hat{s}_n$ is given by

$$e_n = s_n - \hat{s}_n = s_n + \sum_{k=1}^{p} a_k s_{n-k} \tag{3-5}$$

The quantity e_n is also known as the residual. In the method of least squares the parameters $\{a_k\}$ are obtained as a result of the minimization of the expected value or mean of the error squared term, $Ep = \mathcal{E}(e_n^2)$, with respect to each of the parameters. Ep is the minimum mean square prediction error, averaged over all n, and is represented by

$$Ep = \mathcal{E}(e_n^2) = \sum_{n=1}^{\infty} \left[s_n + \sum_{k=1}^{p} a_k \; s_{n-k} \right]^2 \tag{3-6}$$

For any definition of the signal s_n, a set of equations with a set of unknowns can be solved for the predictor coefficients which minimize Ep.

There are two distinct methods for the estimation of these parameters, namely the autocorrelation method and the covariance method. Both methods are clearly described by Makhoul [Ref. 6:pp. 126-127]. Since the autocorrelation method is the preferred method, only that method will be summarized here.

a. Autocorrelation Method

Here we assume that the error Ep is minimized over an infinite duration. Since

$$R(i) = \sum_{n=-\infty}^{+\infty} s_n \, s_{n+i} \qquad (3\text{-}7)$$

is the autocorrelation function of the signal s_n, Equation 3-6 reduces to

$$Ep = R(0) + \sum_{k=1}^{p} a_k \, R(k) \qquad (3\text{-}8)$$

where R(0) is the total energy of the input signal and R(k) is the autocorrelation matrix of the input signal (see Figure 3).

$$\begin{bmatrix} S_0S_1 & S_1S_2 & S_2S_3 & \cdots & S_{p-1}S_p \\ S_1S_2 & S_0S_1 & S_1S_2 & \cdots & S_{p-2}S_{p-1} \\ S_2S_3 & S_1S_2 & S_0S_1 & \cdots & S_{p-3}S_{p-2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ S_{p-1}S_p & S_{p-2}S_{p-1} & S_{p-3}S_{p-2} & \cdots & S_0S_1 \end{bmatrix} \equiv \begin{bmatrix} R_{0,1} & R_{1,2} & R_{2,3} & \cdots & R_{p-1,p} \\ R_{1,2} & R_{0,1} & R_{1,2} & \cdots & R_{p-2,p-1} \\ R_{2,3} & R_{1,2} & R_{0,1} & \cdots & R_{p-3,p-2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ R_{p-1,p} & R_{p-2,p-1} & R_{p-3,p-2} & \cdots & R_{0,1} \end{bmatrix}$$

Autocorrelation Matrix

Figure 3.

It is a symmetric toeplitz matrix (a toeplitz matrix is one in which all the elements along the diagonal are equal). Since the signal s_n is known over only a finite interval, one popular method to control the size of the

toeplitz matrix is to multiply the signal s_n by a window function w_n. This yields a slightly different signal s'_n, which is zero outside the finite interval.

In any case, the autocorrelation matrix is the means for solving several of the linear predictive coefficients needed to analyze and synthesize speech. The following chapter discusses, in greater depth, what those coefficients are and how they are obtained.

IV. LINEAR PREDICTION OF SPEECH

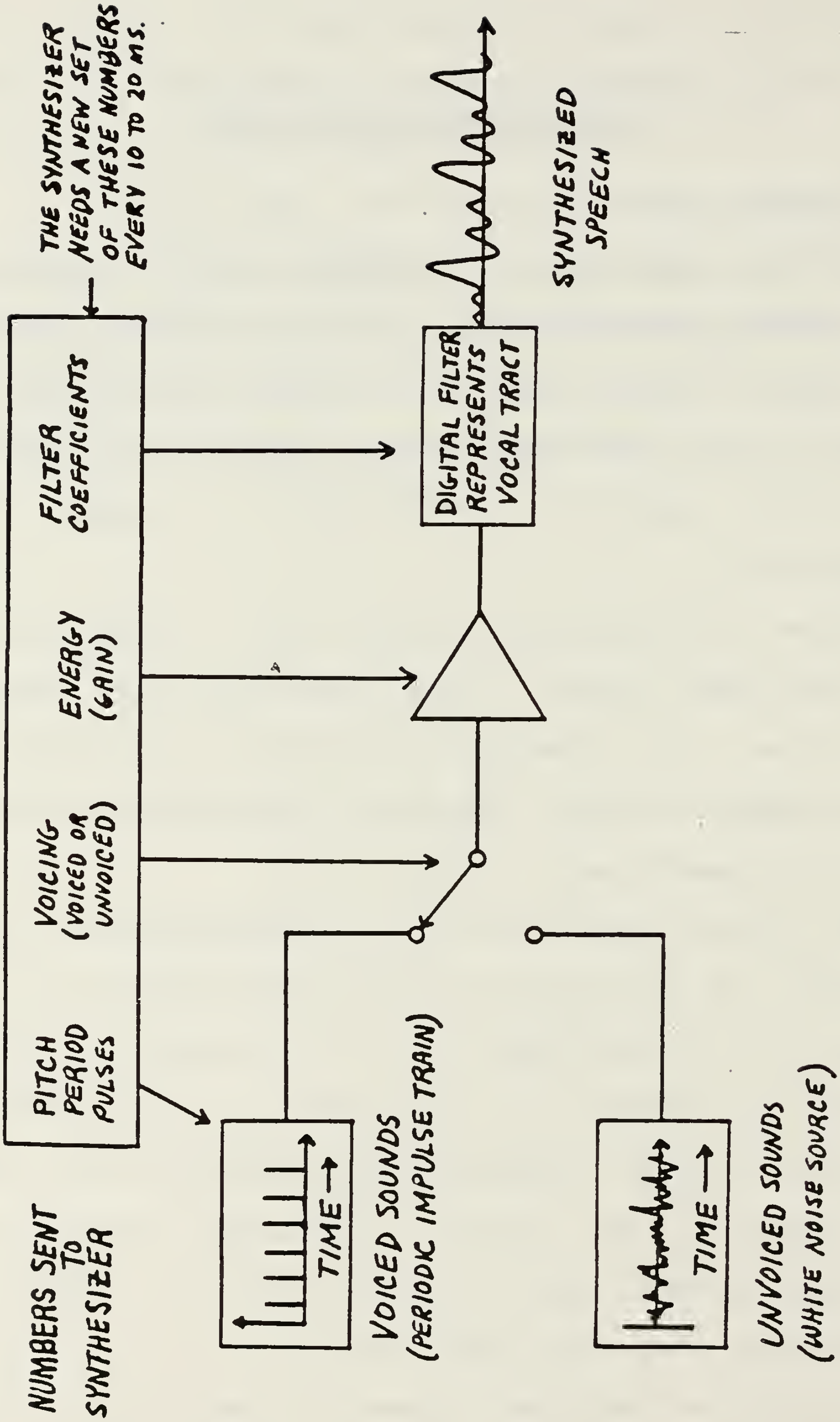
A. INTRODUCTION

As mentioned earlier, there are several ingredients or time-varying parameters that are needed to generate speech. When using linear predictive coding techniques, three ingredients are essential: gain or energy, pitch period, and the filter reflection coefficients or spectral envelope parameters.

Figure 4 illustrates the fact that, depending on the specified frame length, these ingredients must change every 10 to 20 ms. On a frame-by-frame basis the incomming signal is processed to obtain the gain, the pitch period and the reflection coefficients k1, k2,...,kN.

The pitch period and the gain parameters are used to construct an excitation function for production of either voiced or unvoiced speech. This driving or excitation function is input to a filter which is configured by the spectral envelope parameters determined from the analysis. The output is one frame of synthetic speech, and by stringing several frames of speech together, audible sounds are produced [Ref. 7:pp. 337 - 345].

Analysis of the speech signal is done by calculating the LPC model parameters for each 10 ms time frame. This chapter will discuss these essential parameters.



A LPC Model of the Human Voice [Ref. 7:p. 338].

Figure 4.

B. LPC ENCODING PARAMETERS

1. Voiced / Unvoiced Decision Making

Some sounds require the vibrations induced by the vocal cords, while others do not. Voiced sounds represent those that require an excitation from the vocal cords or lips. Unvoiced sounds are generated by a steady flow of air as in the case of 's' or 'f'. A decision must be made in order to properly excite the digital filter to produce the desired sounds.

According to Atal [Ref. 5:p. 280] the voiced/unvoiced decision is based on the ratio of the mean-squared value of the speech samples to the mean-squared value of the prediction error samples. This ratio is considerably smaller for unvoiced speech sounds than for voiced speech sounds. Typically, this ratio is a factor of 10.

$$\text{Voiced Decision:} \quad E[s_n] \geq 10\, E[e_n]$$

$$\text{Unvoiced Decision:} \quad E[s_n] < 10\, E[e_n]$$

This decision will determine whether to excite the digital filter with an impulse function or white noise, each having a particular gain or energy.

2. Gain Computation

In explaining the least squares method of linear prediction we assumed that the input was unknown.

Equation 3-5 can be rewritten as

$$s_n = - \sum_{k=1}^{p} a_k s_{n-k} + e_n \tag{4-1}$$

Comparing Equations 3-2 and 4-1 we see that the only input signal u_n that will result in the signal s_n as output is that where $Gu_n = e_n$. That is, the input signal is proportional to the error signal. For any other input the output will be different than s_n. Therefore the energy of the input signal must be equal to the energy of the output signal s_n.

Since the filter H(z) is fixed, it is clear from the above that the total energy in the input signal Gu_n must equal the total energy in the error signal, which is given by Ep. Again, Makhoul [Ref. 6:p. 128] is the primary source for this information and he provides additional mathematical background in determining the resultant gain equation

$$G^2 = E_p = R(0) + \sum_{k=1}^{p} a_k R(k) \tag{4-2}$$

where G^2 is the total energy in the input and R(k) is, again, the autocorrelation matrix.

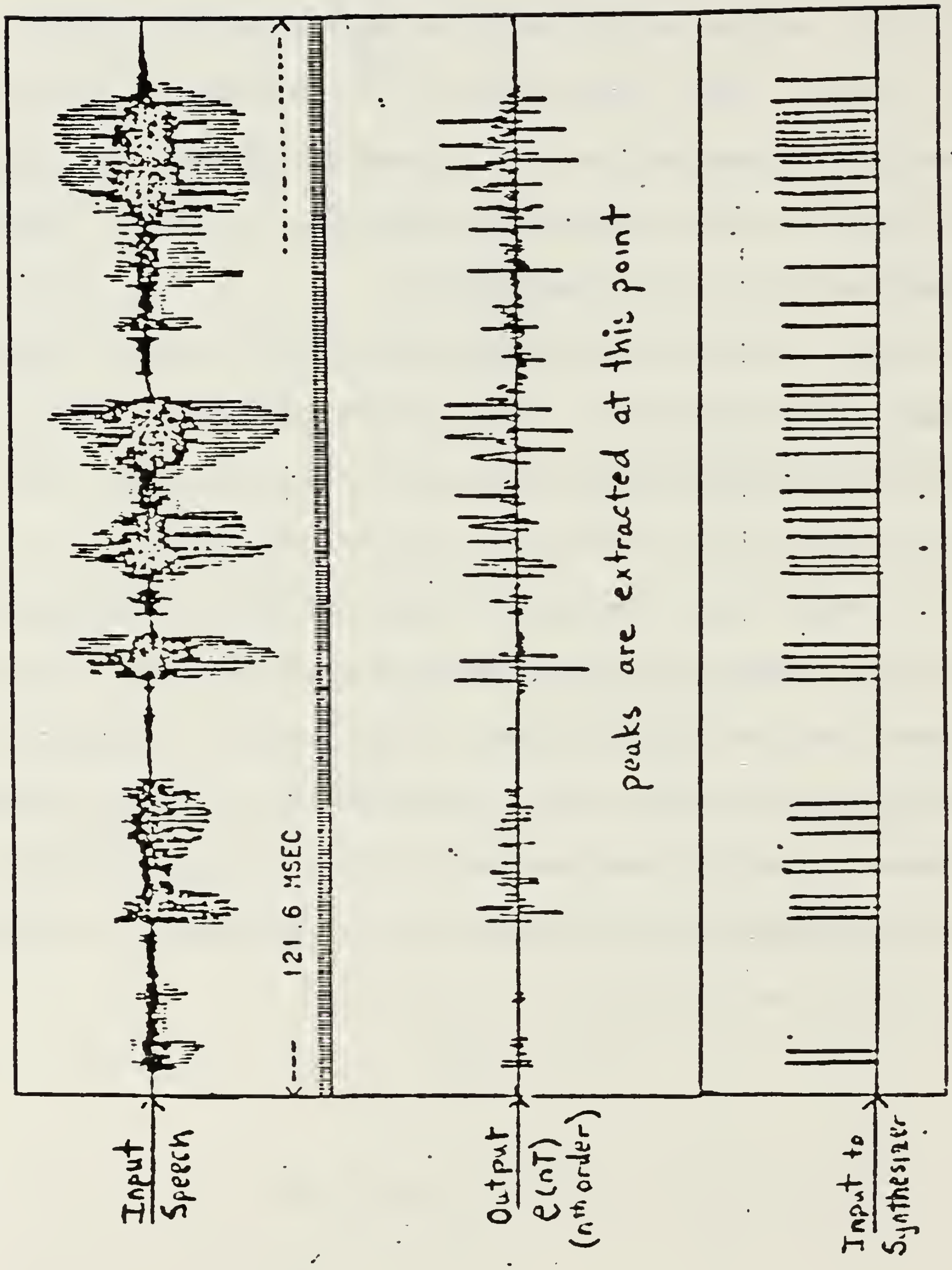
The classification of a sound as voiced or unvoiced determines the input to the filter H(z). However if the input Gu_n is white noise or a series of impulses, the gain is calculated from the same equation.

3. Pitch Period

The period of time that elapses between each excitation pulse is referred to as the pitch period. Atal [Ref. 5:p. 279] describes two different methods for determining pitch period. His second method is summarized here since it is based on the linear predictive representation of the speech wave.

In this method, except for a sample at the beginning of each pitch period, every sample of the voiced speech waveform can be predicted from the past values . The method of determining pitch period is relatively simple.

Once the prediction error of the speech signal is determined through linear predictive processing, the largest or peak values are noted, (Figure 5). These points determine the times that excitation pulses should be initiated from the excitation source. This simple peak-picking procedure was found to be effective in determining pitch period as developed in Reference 7.



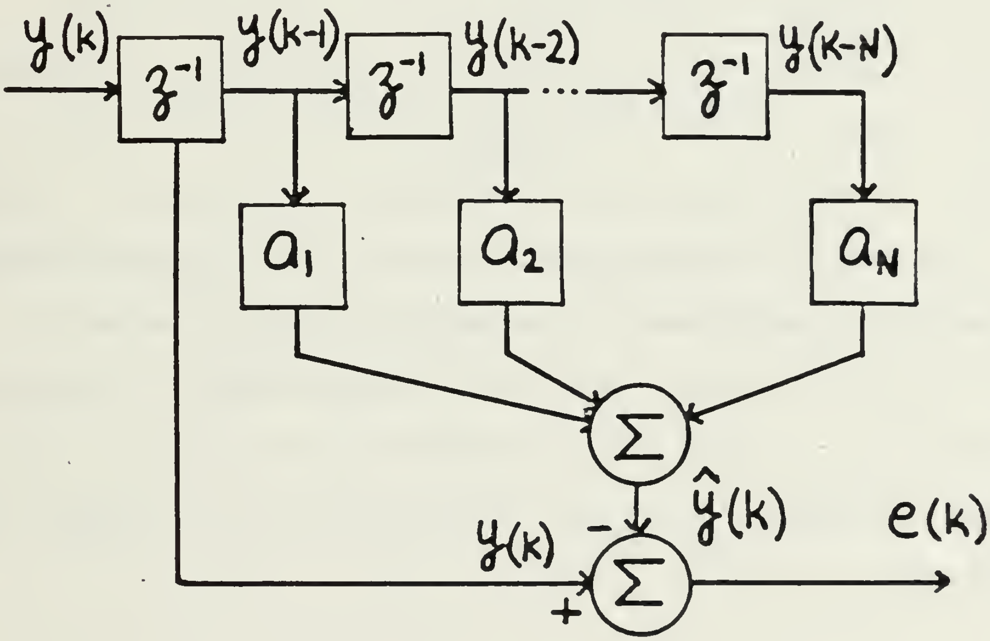
Pitch Period Estimation Using Peak Picking

Figure 5.

4. Reflection Coefficients

Earlier it was mentioned that the reflection coefficients determined using LPC are directly related to the polynomial coefficients of an all pole model. This section will show the relationship between them and illustrate how the reflection coefficients are determined.

Recall that we are looking for an estimated output which is the weighted sum of past system outputs (see Eqns. 3-4 and 3-5). The autoregressive (AR) model in Figure 6 illustrates this process.

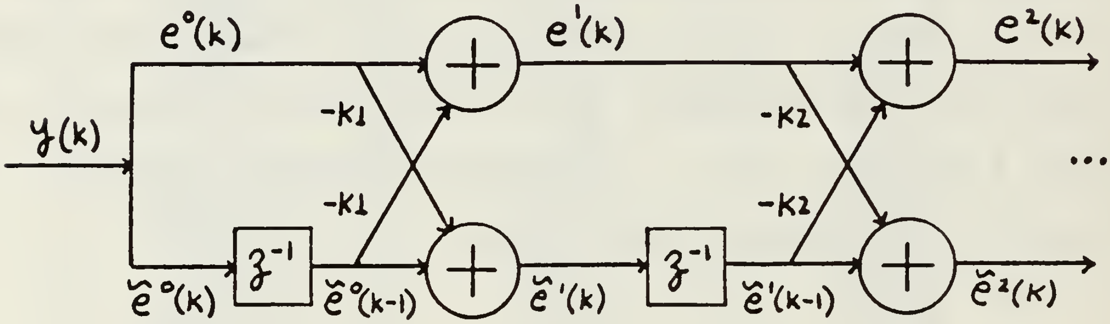


Autoregressive Model

Figure 6.

The goal of LPC is to adjust the a_k's to minimize Ep. Achieving it involves solution of a linear system of

equations, using Levinson's algorithm, and leads to the lattice structure AR model we are most interested in (see Figure 7). The mathematical development for this may be found in Parker [Ref. 9:pp. 110-112].



Lattice Structure Analysis Model

Figure 7.

Lattice structuring requires the determination of reflection coefficients, hereafter referred to as K. The K's of an n-th order Lattice filter transfer function are related to the polynomial coefficients of an nth order AR filter transfer function through the following matrix equation:

$$\underline{a}^{(N+1)} = \begin{bmatrix} a^{(N)} \\ \hdashline 0 \end{bmatrix} + K^{(N+1)} \begin{bmatrix} -a^{(N)} \\ \hdashline 1 \end{bmatrix} \qquad (4\text{-}3)$$

where

$$K^{(N+1)} = \frac{Ryy^{(N+1)} - \tilde{\underline{r}}yy^{T(N)} \cdot \underline{a}_k^{(N)}}{Ryy(0) - \underline{a}_k^{T(N)} \cdot \underline{r}yy} \qquad (4\text{-}4)$$

The matrix $\underline{r}yy$ is the last column of the Ryy autocorrelation matrix mentioned earlier. The notation has been slightly altered from Parker's presentation [Ref. 9:p. 112] to be consistent with the preceding chapters of this development.

Equations 4-3 and 4-4 have been included in this presentation to show how the polynomial coefficients (a_k's) are related tho the reflection coefficients (K's). However, there is an easier and more direct method towards determining K's. A brief development is presented here.

Working in the Z-domain, we know that the transfer function of the AR model is

$$A^{(N)}(z) = 1 - \sum_{i=1}^{N} a_i^{(N)} z^{-N} = 1 - a_1 z^{-1} - \ldots - a_N z^{-N} \qquad (4\text{-}5)$$

and

$$\breve{A}^{(N)}(z) = z^{-N} A^{(N)}(z^{-1}) = z^{-N}(1 - a_1^{(N)} z - a_2^{(N)} z^2 - \ldots - a_N^{(N)} z^N) \qquad (4\text{-}6)$$

where $\tilde{A}(z)$ is A(z) in reverse order.

Combining and reforming in matrix form, yields

$$A(z) = 1 - [z^{-1} z^{-2} \ldots z^{-N-1}] \begin{bmatrix} a^{(N)} \\ \phi \end{bmatrix} - K^{(N+1)} [z^{-1} z^{-2} \ldots z^{-N-1}] \begin{bmatrix} \breve{a}^{(N)} \\ 1 \end{bmatrix} \quad (4\text{-}7)$$

or more simply

$$A^{(N+1)}(z) = A^{(N)}(z) - K^{(N+1)} z^{-1} \breve{A}^{(N)}(z) \quad (4\text{-}8)$$

and

$$\breve{A}^{(N)}(z) = z^{-1} \breve{A}^{(N)}(z) - K^{(N+1)} A^{(N)}(z) \quad (4\text{-}9)$$

Writing Equation 3-5 in the Z-domain yields

$$E^{N}(z) = A^{N}(z) \; S(z) \quad (4\text{-}10)$$

Combining 4-10 with 4-8 and 4-9 and returning to the time domain, yields the following error equations.

$$e^{(N+1)}(k) = e^{N}(k) + K^{(N)} \hat{e}^{(N-1)}(k-1) \quad (4\text{-}11)$$

and

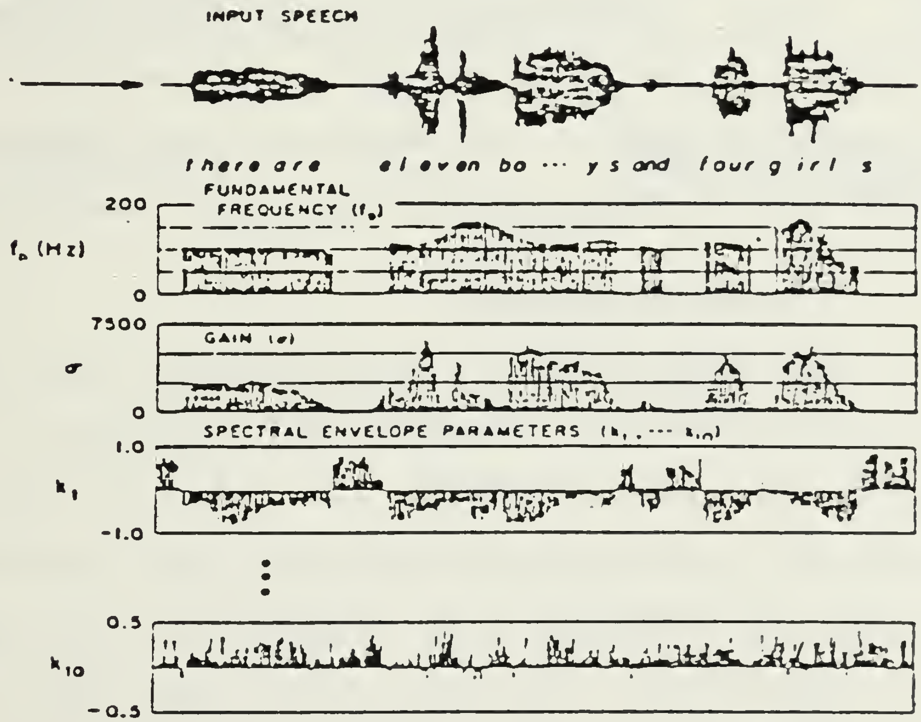
$$\hat{e}^{(N)}(k) = \breve{e}^{(N-1)}(k-1) - K^{(N)} e^{(N-1)}(k) \quad (4\text{-}12)$$

where $e^{(N+1)}(k)$ is the forward difference error, and $\tilde{e}^{(N)}(k)$ is the backwards difference error. Equations 4-11 and 4-12

correspond to the lattice implementation in Figure 7. They have been used to determine the K's of a 12th order model in the sine wave and speech experiments which follow.

The order of the filter is simply determined by assigning N. For speech, anywhere from a 6th to a 12th order model has been found to be sufficient.

The reflection coefficients are determined every 10 to 20 milli-seconds and when lined up side by side appear to present a spectral envelope, (Figure 8).



Display of Analysis/Synthesis Parameters [Ref. 10:p. 16].

Figure 8.

Determining the reflection coefficients, in any case, is a straight forward calculation which is an attractive

feature of LPC. It is the pattern these K's may produce in our experiment that we will be most interested in.

5. Spectral Analysis

A convenient way to portray the frequency content of speech is through the determination of formant frequencies. Formant frequencies are the most prominent frequencies present in a speech waveform.

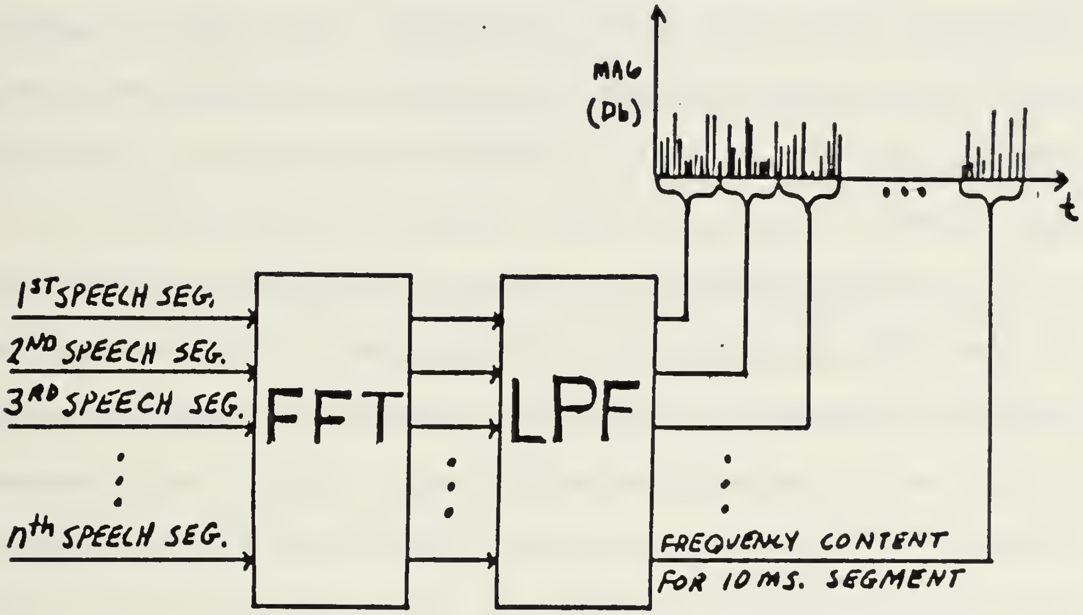
Formant frequencies are not required to produce LPC synthesized speech. In other words, given the voiced decision, gain, pitch period, and the reflection coefficients, one has enough information to reconstruct the speech wave form. However, the determination of the formant frequencies aids us in depicting a frequency transposition.

a. Formant Frequencies

The complex roots of the denominator polynomial are the complex formants (bandwidths and frequencies) used to approximate the speech signal. The coefficients, a_k ,of the denominator polynomial are obtained from time-domain calculations on samples of a short segment of the speech waveform; namely $\{s_n\} = \{s_1 , s_2 ,...s_N\}$, where $N>>p$. Here N is the number of samples, and p is the order of the polynomial [Ref. 11:pp. 364-366].

Under the assumption that the waveform samples, s_n , are samples of a random gaussian process, the entire speech sample is broken up into an equal number of

samples which we will refer to as segments, (Figure 9). Each segment is processed using the Fast Fourier Transform (FFT) and then low pass filtered if desired.



Flow Chart for Obtaining the Spectral Content
of One Complete Utterance

Figure 9.

The output of each segment contains the spectral content of that segment. Each segment is sequenced together to yield a time-varying frequency content profile of the entire utterence with each segment containing its particular frequency content. The formant frequencies are the most prevalent, or peak, frequencies found in the speech wave form.

C. SPEECH SYNTHESIS

A speech signal is synthesized by using the same parameters determined with LPC analysis. A block diagram of a speech synthesizer was shown in Figure 4. The control parameters supplied to the synthesizer are the pitch period, a binary voiced or unvoiced parameter, the rms value of the speech samples or gain, and the predictor or reflection coefficients.

The pulse generator produces a pulse of unit amplitude at the beginning of each pitch period. The white noise generator produces uncorrelated uniformly distributed random samples with standard deviation equal to 1 at each sampling instant. The selection between the pulse generator and the white noise generator is made by the voiced-unvoiced switch. The synthesizer control parameters are reset to their new values at the beginning of every pitch period for voiced speech and once every 10 msec for unvoiced speech.

The amplitude of the excitation signal is adjusted by the amplifier G. The linearly predicted value s_n of the speech signal is combined with the excitation signal u_n to form the n-th sample of the synthesized speech signal. The signal is finally low-pass filtered to provide the continuous speech wave $\{s_n\}$. Atal [Ref. 5:p. 280] provides the mathematical development needed to synthesize these parameters. A mathematical discussion will not be pursued further here.

V. DIGITAL FREQUENCY TRANSPOSITION

A. INTRODUCTION

The object of this research was to determine an algorithm that will digitally transpose speech using linear predictive coding. In this chapter, Hall's research [Ref. 3] will be briefly discussed and summarized. A new theory will then be postulated and a simple experiment using pure sine waves will be presented to test the credibility of the theory. Keep in mind that the real test will be the actual processing of speech, this section simply sets the scene for further study.

B. POLE SHIFTING IN THE Z-PLANE

Only the highlights and summary of Hall's thesis will be presented here. His goal was to change the pole locations before reconstruction (of the sampled speech signal) to produce the output voice with different pitch and format frequencies while retaining a natural sound and the same information [Ref. 3:p 47].

The autoregressive vocal tract transfer function used in his research is represented by Equation 5-1.

$$H(z) = \frac{1}{1 - 2e^{-2\pi(BW)T_S}\cos(2\pi FT_S)z^{-1} + e^{-4\pi(BW)T_S}z^{-2}} \qquad (5\text{-}1)$$

where F is the center frequency of the formant, and BW is the bandwidth of the formant. The pole locations associated with this transfer function are:

$$z = A\,e^{\pm j\theta}$$

Converting Equation 5-1 into polar form produces Equation 5-2.

$$H(z) = \frac{1}{1 - 2A\cos\theta z^{-1} + A^{2}z^{-2}} \qquad (\text{Eqn } 5\text{-}2)$$

Through several mathematical manipulations and solving for A and θ, the following relationships for F and BW are determined:

$$F = \theta \,/\, 2\pi T \qquad (5\text{-}3)$$

$$BW = (-\ln A)\,/\, 2\pi T \qquad (5\text{-}4)$$

where $A = e^{-2\pi(BW)T}$ and $\theta = 2\pi FT$

Assuming that a linear relationship exists between F (the original frequency) and F′ (the modified frequency), several general expressions are stated to illustrate the

underlying modification to the pole locations. Note that the following equations are all linear relationships.

$$F' = \gamma F \tag{5-5}$$

$$BW' = \alpha BW \tag{5-6}$$

$$\theta' = \gamma \theta \tag{5-7}$$

$$A' = A^{\alpha} \tag{5-8}$$

The most important consideration for producing these relationships is guaranteeing that no unstable poles will be created by shifting them outside the unit circle. For more of the specifics on Hall's development see Reference 3, pages 49 and 50.

Two experiments are illustrated in Hall's thesis. They are:

1. Pitch was reduced by a factor of .58 and the formant frequencies reduced by .88 for voiced speech.

2. The same modification was done for a segment of unvoiced speech.

Hall concluded that upon completion of the process most listeners agreed that, although the input speech was female, the modified output speech sounded typically male. It was also noted that although the audio output was somewhat lacking in quality, it was intelligible [Ref 3:p. 73]. The tapes which recorded that audio output are no longer available for subjective evaluation.

Linear predictive coding is a means to an end for Hall. He modifies the the variables mentioned (F,BW,O,A), and processes the speech with LPC computer programs. This conversion between an autoregressive vocal track model and a LPC model (implemented most easily by a lattice filter configuration) is possible through Equations (4-3) and (4-4).

The mathematics are simple. What is most important here is that the relationship between the two different representations of speech, the AR model and the LPC model, are closely associated with one another. To calculate one, in a sense, is to calculate the other.

C. A NEW PROPOSITION

1. Statement of Theory

As mentioned earlier, LPC techniques can serve as a tool for modifying the acoustic properties of the speech signal. This thesis postulates that a linear relationship exists between the reflection coefficients, which determine the spectral envelope of the speech wave form, and the frequency content of that wave form. If this relationship exists and the linear relationship is determined, then by selectively modifying the reflection coefficients, the frequency content will also be modified.

Is there a linear relationship between the reflection coefficients (K's) and frequency content? The

first step in our proof is to analyze the most simplified case. Since speech is often represented as a combination of many different frequencies, the simplest case would be to analyze a fixed frequency sine wave. If the results turn out to be negative, then exploring the more complex case (speech) would probably be futile.

2. Sine Wave Experiment

At any given frequency a pure sine wave may be considered a continuous energy and amplitude signal which will generate an audible pitch when it is within the 200 Hz to 15 kHz audible range. When dealing with normal speech wave forms, the audible pitch range is somewhere between 200 Hz and 5 kHz.

A computer program was written in Fortran [App. A], for use on the IBM 3033 to produce a sine wave for further analysis. The resultant sine wave could be sampled at any desired rate and the frequency of the wave could be incremented to satisfy the range requirements of 200 Hz - 5 kHz.

Once the sampling rate was determined and the sine wave frequency set, the reflection coefficients were calculated for a 10ms time frame, stored in a holding file and plotted to determine if a relationship exsists between frequency and the nth order K's.

To determine 12 reflection coefficients (K's) for each frequency, Equations 4-11 and 4-12 were used. Additional runs were also made to determine the affect of noise on the outcome. The results were promising.

3. Sine Wave Experimental Results

Appendixes B and C illustrate the apparent linear relationship that exsists between frequency and the LPC nth order K's in a noiseless environment. Appendixes D and E illustrate that same relationship in a noise environment (S:N = 10:1).

It would appear that a linear relationship does exist between the different frequencies of a sine wave. Noise on the other hand changes that linear relationship. Noise addition seems to affect K7 through K12 much more than K1 through K6.

Considering the mathematics involved in calculating K, these observations are reasonable. Since the later K's are affected most by small changes in the input signal, addition of noise will affect them more drastically than the earlier stages .

Though these observations are promising, they are by no means conclusive. If no correlation between the K's and frequency existed, another scheme would have had to be considered. Nevertheless, speech is the more complicated signal that we consider in the next two sections.

VI. SPEECH PROCESSING EXPERIMENT

A. INTRODUCTION

Now that the fundamentals of linear predictive coding have been presented and a theory of frequency transposition proposed, it is necessary to work directly with speech itself. To obtain the information we are seeking, the correlation between reflection coefficients and frequency content, speech samples must be demonstrated.

Documentation concerning the data acquisition system used in this research to obtain speech samples is provided as Appendix F. This chapter discusses the data itself, and the processing of it.

B. VOICED/UNVOICED PHRASES

Three phrases were chosen for their voiced and unvoiced characteristics as described in Chapters 2 and 4. They are:

1) "READY"
2) "SO WHAT"
3) "SNEEZE"

Each phrase was repeated at a different pitch and to make things simple, the musical scale was picked to help harmonize a change in pitch with some type of reference. In other words, "READY" was first spoken in the middle-C range,

and then in the D range, until it was finally spoken in the high-C range.

This procedure yielded eight different pitches for each of the three phrases. One male speaker provided the data for all three phrases. Additionally the period remained constant for each pitch and their individual utterances. For a graphical representation of the selected speech utterances, refer to Appendices G, H, and I.

Each phrase was chosen for content and can be classified as voiced, unvoiced, or a combination of both. "READY" is strictly a voiced word, whereas "SO WHAT" and "SNEEZE" are a combination of voiced and unvoiced segments. The S,WH, and T sounds in "SO WHAT" will be our unvoiced example, and "SNEEZE" will be the combined example as the data is analyzed.

C. DATA PROCESSING

This section discusses the techniques utilized to analyze the data and the observations made.

1. Speech Data

The raw speech data was edited and displayed using a generic display program. The data is 8 bit information with a maximum range of 256 equally spaced values. The resolution of each utterance varied with the pitch. The lower frequencies tended to have less gain or energy and

therefore did not use all the 256 range values available. A summary of the ranges is provided in Appendix J.

The periods of each phrase were different. The differences between the same utterance at different pitches varied as much as 20 msec. A short summary of the average periods are given in Table 1.

TABLE 1.

UTTERANCE "XXX"	PERIOD sec.	NO.SEGMENTS N	NO.DATA PTS./SEG (10 msec SEG)
"READY"	.30	30	100
"SO WHAT"	.40	40	100
"SNEEZE"	.38	38	100

The sampling rate was 10 kHz for all of the utterances, so the number of data points in each 10 msec segment is 100.

2. Determining Reflection Coefficients

Once the starting point is determined for each utterance, the reflection coefficients are calculated for 10 msec segments of speech [App. K]. Successive segments are analyzed to yield their respective reflection coefficients

using Equations 4-11 and 4-12, as were the sine wave calculations.

Reflection coefficients K1 through K6 were plotted for each of the 24 utterances and several of the resultant curves are included as Appendix L.

a. Trend Analysis

A graphical trend analysis of the plotted data was undertaken to detect any obvious patterns. The details of that analysis is included as Appendix M. However, a summary of those observations leads us to the conclusion that there were not any trends of any significance noted as a function of pitch.

b. Graphical Correlation

One graph was held stationary as a reference and the others were passed over it to see if there was any obvious match ups. There is nothing more elaborate to report than that no correlation was noted between them. Even though at times there were 2 or 3 points which matched up, the other 28, 36, or 38 points did not. Also there seemed to be no distinction between voiced and unvoiced portions of the speech wave. This process leads to the conclusion that the various speech segments are highly uncorrelated.

3. Spectral Analysis of Reflection Coefficient Patterns

It was noted during the trend analysis that the temporal patterns presented by the reflection coefficients seemed periodic. At first it was believed that this could possibly reflect the pseudo-periodic nature of speech or the excitation source.

Spectral analysis was implemented using a Fortran subroutine to compute the FFT of each pattern. The program is included as Appendix N and several examples of the results are provided as Appendix O.

In summary all of the spectra turned out to be relatively flat. This indicates that there are no prominent frequencies within the reflection coefficient sequences.

4. Spectral Analysis for Frequency Content

Spectral analysis to determine the frequency content of each utterance, as described in Chapter 4, would have been useful had a pattern or linear relationship shown up in the observations mentioned.

Since there are no patterns or correlations worth mentioning, exploring the specific frequency content of each utterance would not benefit us. The relative difference between each frequency, or Δf is approximately 32 Hz.

The range of the utterances was chosen to coincide with the musical scale from middle-C to high-C (a 256 Hz difference). Had a relationship been discovered, as

proposed, then a more in-depth spectral analysis of the input speech would have been in order.

D. SUMMARY OF EXPERIMENTAL RESULTS

1. Correlation Between Phrases With Different Pitches

The linear relationship postulated in Chapter 5 should have yielded more obvious results if relationships did exist between identical phrases spoken at different pitches. Three of the four categories mentioned above yielded negative or uncorrelated results.

2. Voiced/Unvoiced Observations

Though there may be other or more sophisticated techniques available to analyze this data, the methods mentioned above were sufficient to show that a voiced phrase was no more correlated than an unvoiced phase.

Since the results were consistently negative or uncorrelated leads us to some conclusions about the actual relationship between frequency content and reflection coefficients.

VII. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

A new theory to transpose frequency was postulated and tested. Initial results, using sine waves, seemed positive and lead to a further study using speech waveforms. The preceding experiment and subsequent analysis of speech showed no apparent correlation between pitch and reflection coefficient values. These results may be attributed to the following reasons.

1. Complexity of Speech

The speech wave form is a very complex combination of gain, excitation, and spectral content. To pick out one particular attribute and analyze it for a particular phenomenon, such as frequency content, may be unrealistic.

Speech has historically been modeled as a combination of sine waves. However, slow progress in the field of speech processing has caused engineers to rethink this point in terms of the physics involved in generating speech. This leads to our next conclusion.

2. Physical/Mathematical Relationship

The experimental results indicate that, in this case, there is no obvious relationship between the physics

(pitch) of speech and the LPC mathematical representation of speech (reflection coefficients).

This observation makes sense since reflection coefficient determination is based on probabilistic methods, error feedback, and random input samples, the resultant output of each lattice stage no longer resembles the original signal. Once the error signal passes through the first stage of the lattice network, its characteristics have been altered as much as 10 percent. Reflection coefficients are therefore a tool for determining predicted error calculations based on past inputs, and not a physical interpretation of the signals content.

Just as engineers are in error when they refer to the pattern that successive reflection coefficients present as its spectral envelope, reflection coefficients do not directly reflect the frequency content of the signal.

3. Periodic/Pseudo-Periodic Differences

Simulation and experimental results show that reflection coefficients work differently with periodic signals (sine wave) than with pseudo-periodic signals (speech).

In calculating the reflection coefficients for a sine wave, the samples of one frequency are changed very slightly from the previous frequency's samples. Therefore the calculated reflection coefficients also change very

slightly. This observation may be useful in the design of an LPC musical synthesizer, where frequency content and adjustment is processed in a controlled environment.

On the other hand, speech behavior is more random than music. It is pseudo-periodic in the sense that complex vibrations are necessary to produce the speech waveform. However, the rate and randomness at which those vibrations change frequencies seems to prevent the reflection coefficients from having any kind of linear relationship with frequency content.

It is therefore the conclusion of this research that the relationship between frequency content of speech and reflection coefficients is sufficiently complex that modifying reflection coefficients in order to transpose pitch will not be practical.

B. RECOMMENDATIONS

The conclusions have stated that there is no linear relationship present between frequency content and reflection coefficients. Recall that the motivation behind this research was based on Hall's research [Ref. 3] concerning pole shifting. Therefore the following actions are recommended if further or more extensive study is desired.

1. Continue Hall's research using LPC as a tool for speech analysis/synthesis, but focusing attention on the shifting of poles and not on the adjustment of reflection coefficients.

2. Use a data acquisition system that yields 12 or 16 bit resolution of the speech samples.

3. Build a larger data base containing speech utterences at different pitch levels and have the speakers be both male and female.

4. Have the ability to match articulation patterns and synchronize points where speech utterences begin and end.

5. Synthesize the input and processed speech to check for intelligibility of the utterences.

6. Use more sophisticated techniques for pattern recognition.

It is believed that the preceding recommendations, if followed, will help substantiate or refute Hall's research as well as the findings of this research. The need for an adequate technique for frequency transposition still exists.

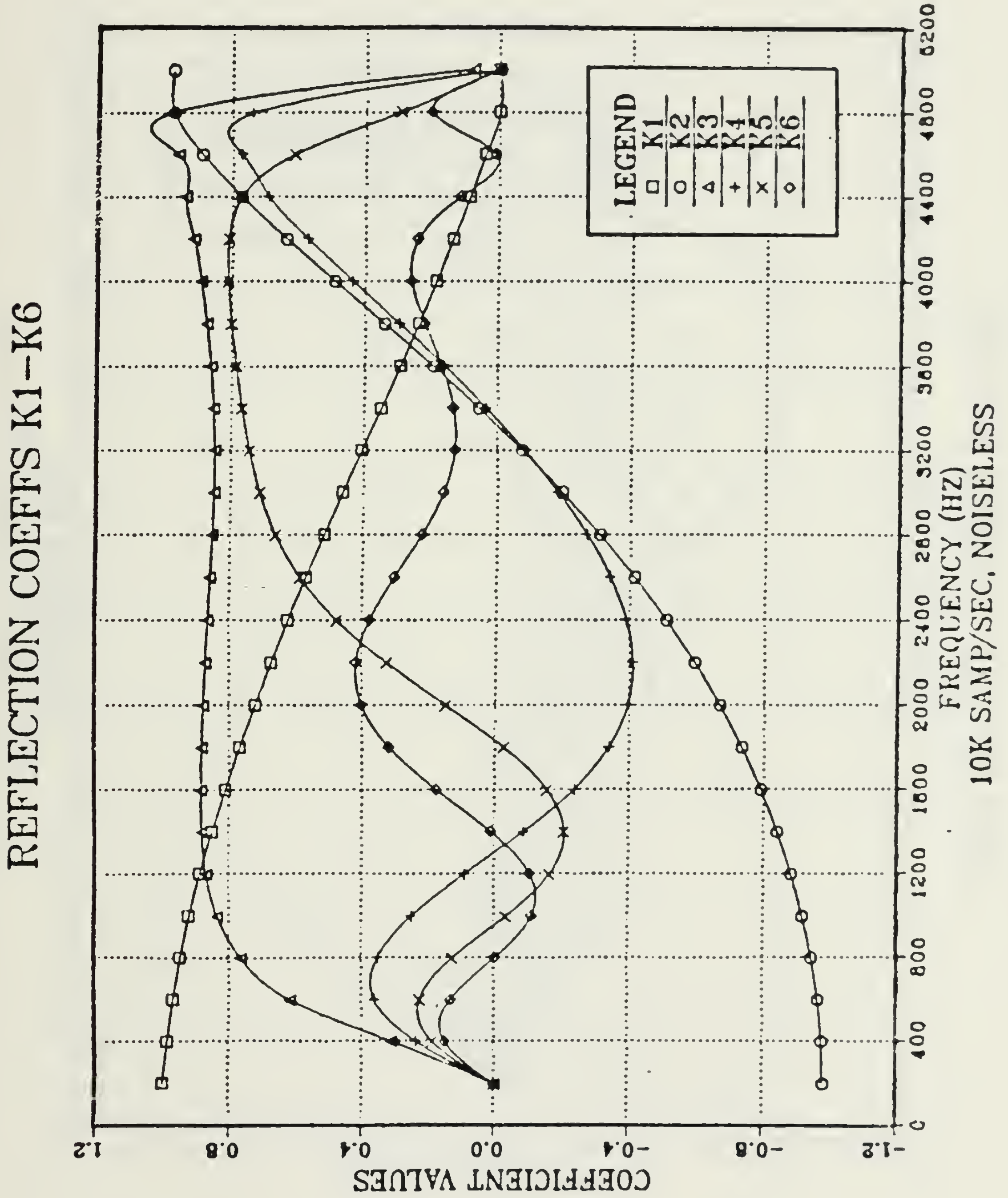
APPENDIX_A - REFLECTION_COEFFICIENT_DETERMINATION_FOR_A FREQUENCY_VARIED_SINEWAVE_PROGRAM

This program determines the reflection coefficients for a 12th order lattice filter model of a variable frequency sine wave.

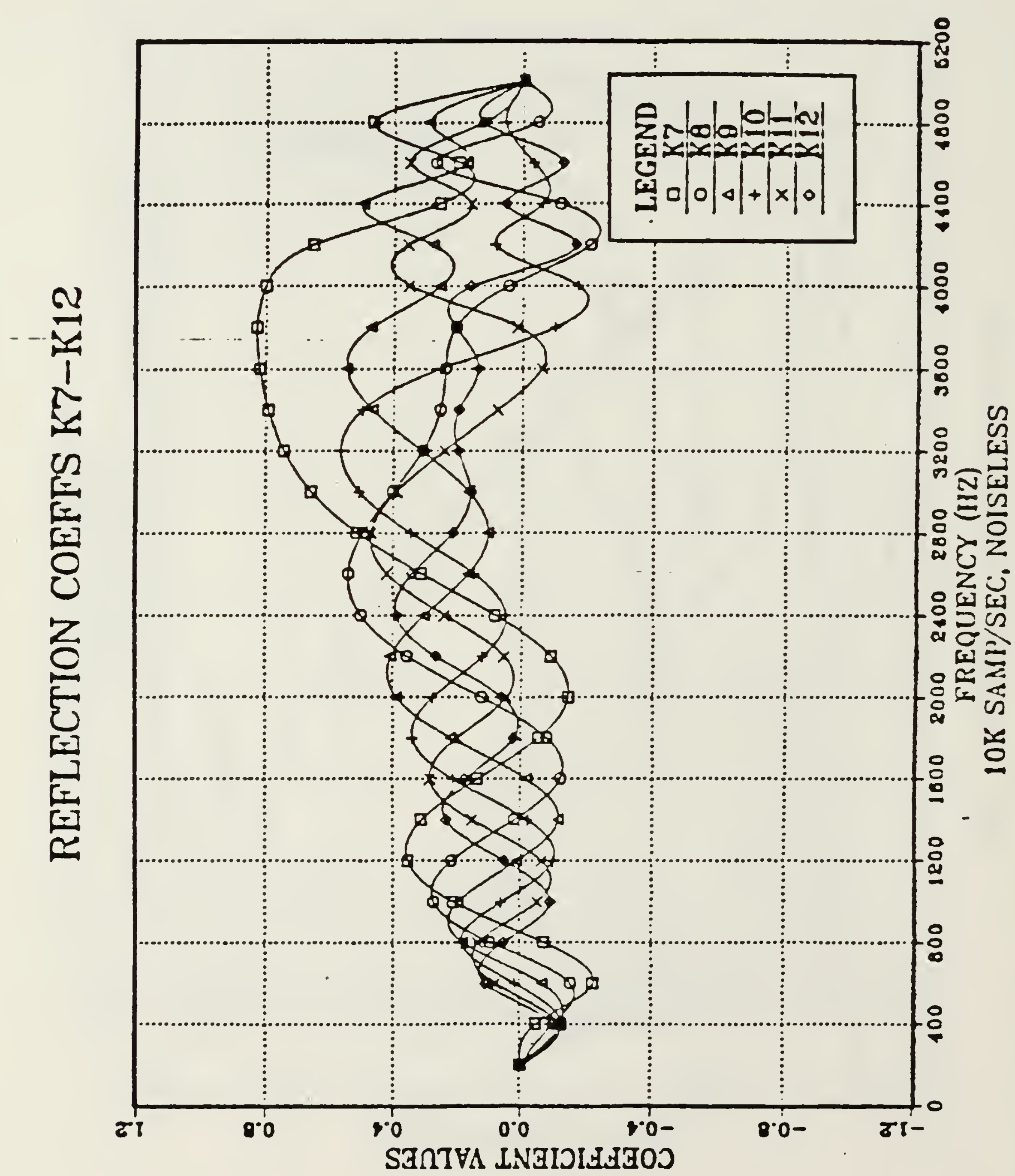
```
FILE: GOTIT    FORTRAN  A1

C*****************************************************************
C
C      DETERMINES 12 REFLECTION COEFFICIENTS OF A LATTICE FILTER
C
C            FOR VARIOUS FREQUENCIES  OF A SINE WAVE
C
C                USING DOUBLE PRECISION MATH
C
C*****************************************************************
C     OS - ORIGINAL SIGNAL
C     RN - RANDOM NOISE
C     T - SAMPLE PERIOD
C     A - AMPLITUDE OF THE SIGNAL
C     K(N) - REFLECTION COEFFICIENTS
C     AC - AUTOCORRELATION MATRIX (1XN)
C     FO - FREQUENCY OF THE SINE WAVE
C     FE - FORWARD ERROR. BE - BACKWARD ERROR
C     SUMN - E<OS(N)*OS(N+1):NUMERATOR
C     SUMD - (E(OS(N))**2):DENOMINATOR
C****************************************************************
      INTEGER N.I.J.L.M.D
      DOUBLE PRECISION OS(1000).SUMN.SUMD.AC(12).T.FO.DB.
     *K(12).FE(12.1000).BE(12.1000).MS(12).A.R.O.
     *YFL.IX.IY.RN1(1000)
      A=10.0
      SUMN=0.0
      SUMD=0.0
      T=10.0E-3
      OS(1)=0.0
      OS(102)=0.0
      DO 5 I=1.35
      F0=200.0*I
C*************************************************************
C     DETERMINE SOME RANDOM NOISE WITH THIS GENERATOR
C*************************************************************
      IX=3789
      DO 44 J=1.101
      CALL RANDU(IX.IY.YFL)
      IX=IY
      RN1(J)=YFL
44    CONTINUE
C******************************************************************
C     OBTAIN THE ORIGINAL SIGNAL ( A SINE WAVE ) WITH OR WITHOUT NOISE
C******************************************************************
      O=2*3.1415927*F0
```

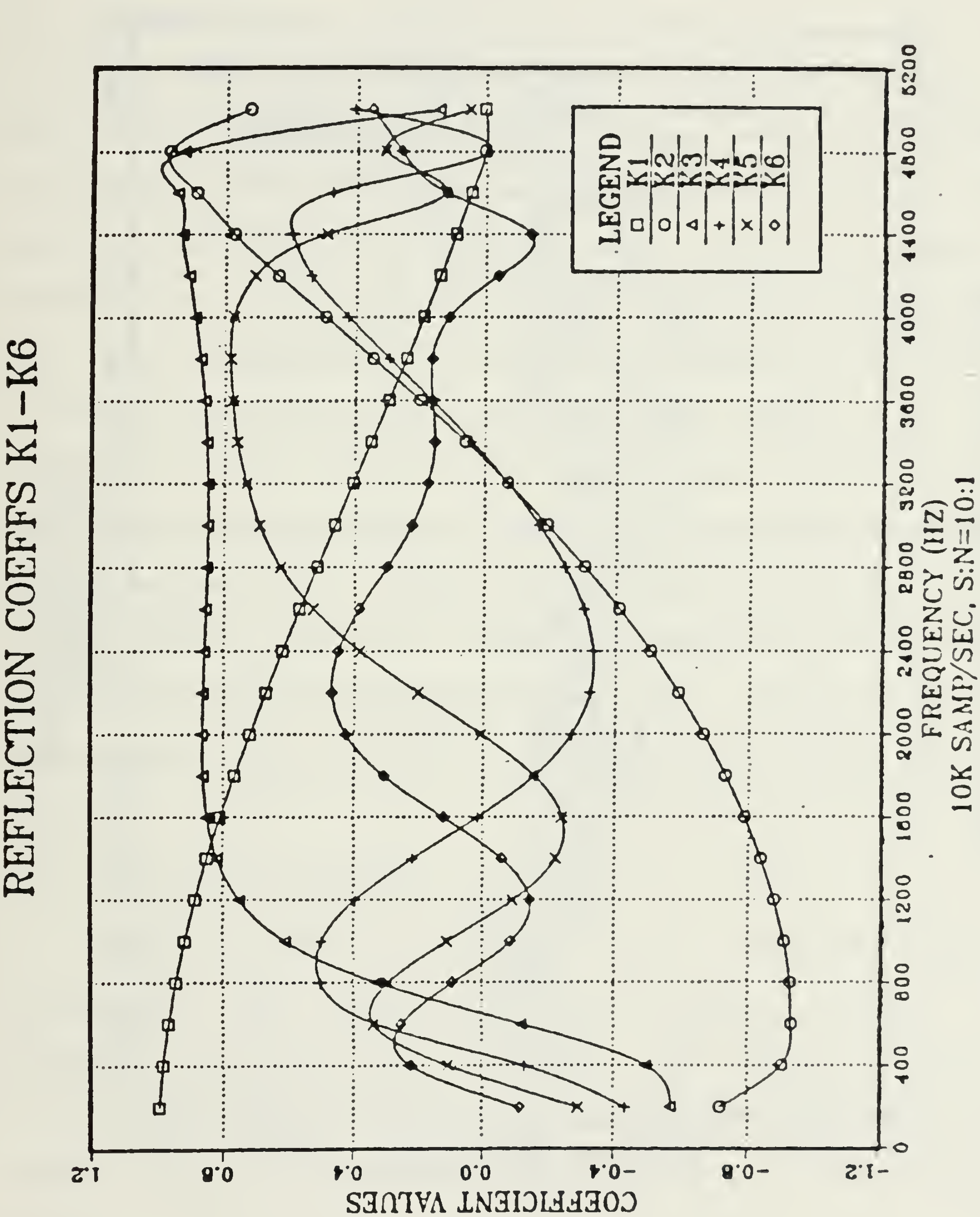
```
      DO 10 J=1,100
      R=((Q*DFLOAT(J)*T)/100)
      OS(J+1)=(A*DSIN(R))+RN1(J+1)
C     OS(J+1)=(A*DSIN(R))
10    CONTINUE
C*************************************************************
C              ANALYSIS MODEL
C*************************************************************
      DO 20 N=1,100
      SUMN=SUMN+(OS(N)*OS(N+1))
      SUMD=SUMD+((OS(N))**2)
20    CONTINUE
      K(1)=SUMN/SUMD
      FE(1,1)=0.0
      BE(1,1)=0.0
      DO 30 J=2,101
      FE(1,J)=OS(J)-(K(1)*OS(J-1))
      BE(1,J)=OS(J-1)-(K(1)*OS(J))
30    CONTINUE
      DO 40 N=2,12
      FE(N,1)=0.0
      BE(N,1)=0.0
      AC(N-1)=0.0
      MS(N-1)=0.0
      DO 50 J=2,101
      AC(N-1)=AC(N-1)+(FE(N-1,J)*BE(N-1,J-1))
      MS(N-1)=MS(N-1)+(FE(N-1,J)**2)
50    CONTINUE
      K(N)=AC(N-1)/MS(N-1)
      DO 60 J=2,101
      FE(N,J)=FE(N-1,J)-(K(N)*BE(N-1,J-1))
      BE(N,J)=BE(N-1,J-1)-(K(N)*FE(N-1,J))
60    CONTINUE
40    CONTINUE
C*************************************************************
      WRITE(4,300)F0,K(1),K(2)
      WRITE(7,300)F0,K(3),K(4)
      WRITE(8,300)F0,K(5),K(6)
      WRITE(9,300)F0,K(7),K(8)
      WRITE(10,300)F0,K(9),K(10)
      WRITE(11,300)F0,K(11),K(12)
C*************************************************************
5     CONTINUE
300   FORMAT(2X,D8.2,5X,D23.16,5X,D23.16)
      STOP
      END
```


REFLECTION COEFFS K1-K6
LEGEND
K1
K2
K3
K4
K5
K6
FREQUENCY (HZ)
10K SAMP/SEC, NOISELESS
COEFFICIENT VALUES
1.2
0.8
0.4
0.0
-0.4
-0.8
-1.2
0
400
800
1200
1600
2000
2400
2800
3200
3600
4000
4400
4800
5200

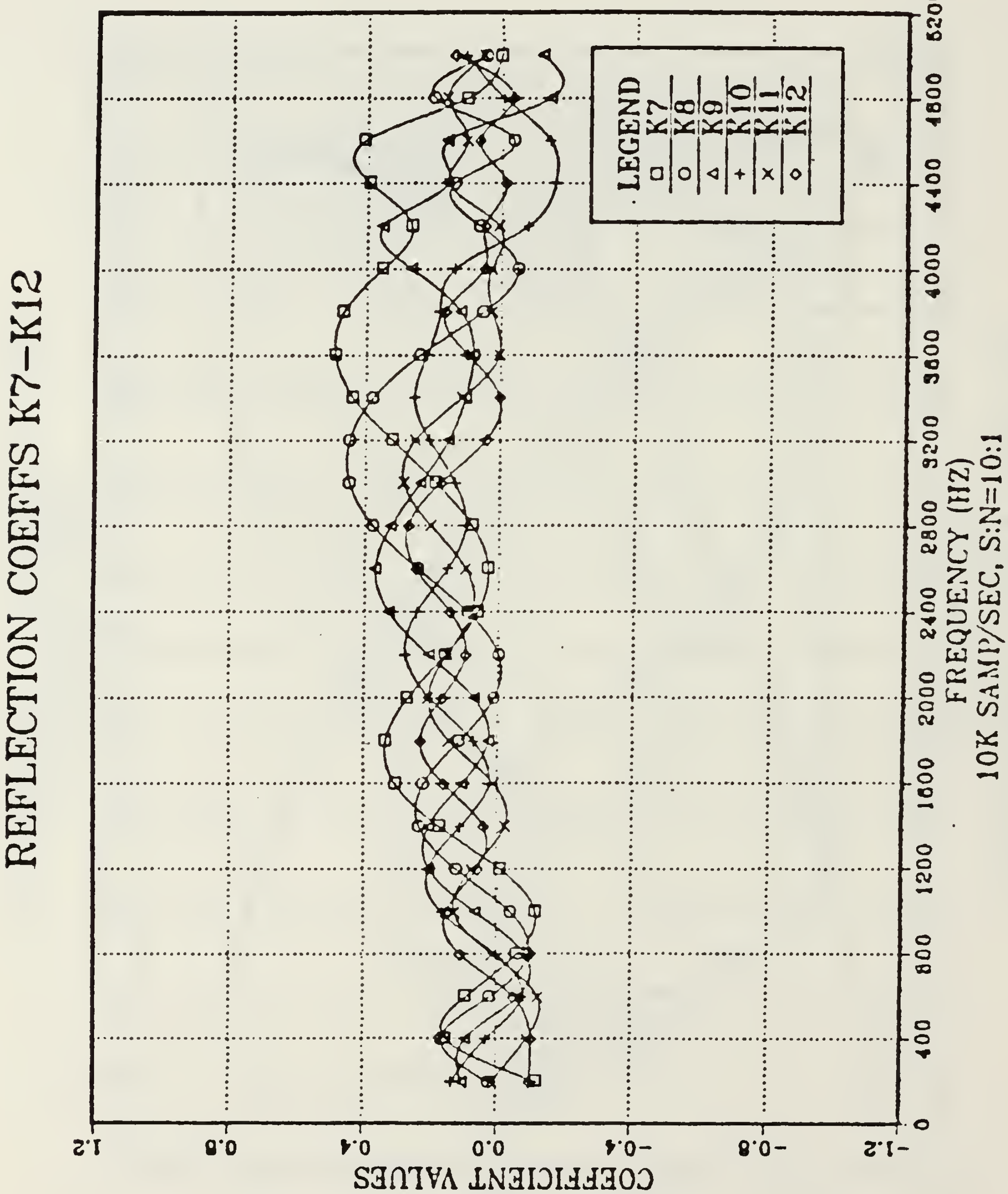
APPENDIX C - REFLECTION COEFFICIENTS K7-K12 (NOISELESS)



APPENDIX D - REFLECTION COEFFICIENTS K1-K6 (S:N=10:1)



APPENDIX E - REFLECTION COEFFICIENTS K7-K12 (S:N=10:1)



APPENDIX F - DATA ACQUISITION SYSTEM

A. INTRODUCTION

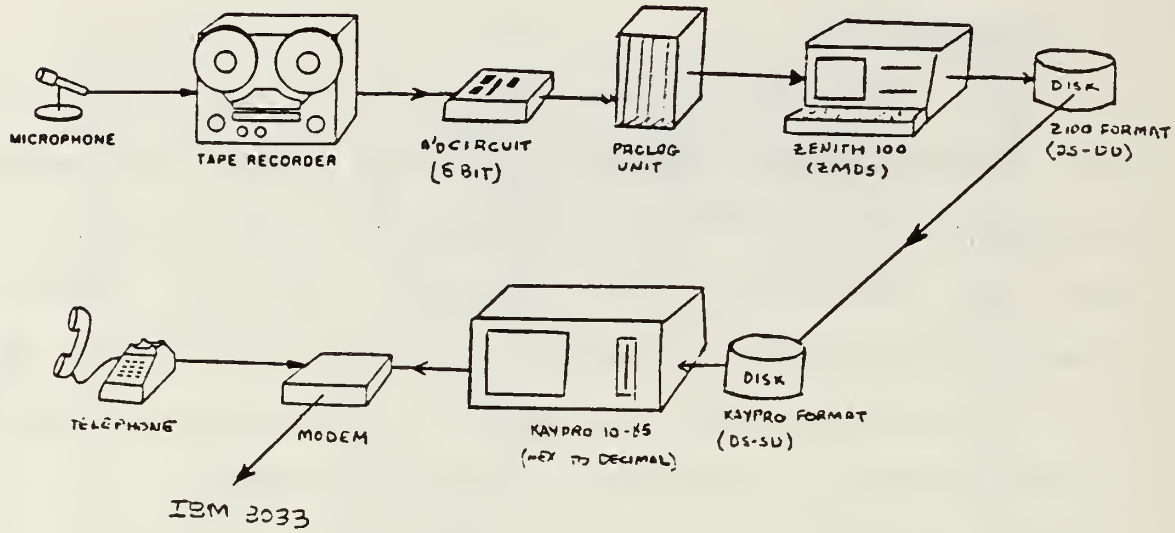
There are a vast number of data acquisition systems on the market today. Though this is the case, the system originally planned for the acquisition of this data, broke down with no hope of timely repair. When all possible alternatives had been explored, it was decided that the only way to accomplish this portion of the research was to build a system capable of obtaining speech data samples.

This section will discuss the system, hardware, and software utilities that were combined to produce the desired data samples. In an effort to provide the novice, as well as the expert, with the information needed to retrace these steps, anything worth documenting, is. Additionally, a bibliography is provided in the main Bibliography of this thesis.

B. EQUIPMENT REQUIREMENTS AND SETUP

Figure 10 shows the experiment. Selected speech utterances were recorded on a 4-channel, 8-track tape recorder and stored for later use. The analog to digital (A/D) circuit was built and driving software written. This circuit was interfaced with the Zenith-100 microcomputer

through the Prolog 7804-Z80A Processor Counter/Timer Card and the 8255 Parallel Peripheral Interface (PPI) microchip.



Data Acquisition 3-Dimensional Flow Chart

Figure 10.

Once the data was captured in the Prolog's 32K buffer, it was uploaded to the Zenith-100, via ZMDS software, and stored in Intel-Hex data files. The files were transferred from the Zenith formatted disk, via an Osborne microcomputer, and placed on Kaypro formatted disks.

A Kaypro 10 microcomputer converted the hexadecimal data into decimal data using Microsoft Basic (MBASIC) software. Edited versions of these files were then transferred to the IBM-3033 main frame computer for data processing.

C. ANALOG TO DIGITAL CIRCUIT

The chip that provides the analog to digital conversion is the AD-570. It provides 8-bit information at sampling rates up to 33K samples/second. For our purposes, the sampling rate was set at 10K since the majority of the frequency content is below 5 kHz.

The circuit diagram [App. F.2] illustrates the interfacing between the 8255 PPI chip and the Host computer. The 8255 coordinates all of the necessary handshaking in driving the AD-570 chip.

It was necessary to amplify the signal prior to entering the AD-570, to obtain full use of the 256 amplitudes available. It was also necessary to provide an adjustable DC-offset to assure a unipolar input (i.e. the middle value had to be adjusted to be level 128 instead of level 0).

Also, the signal was filtered prior to data acquisition, through the use of a Butterworth filter designed with a frequency cutoff of 5 kHz. This helps smooth the data. However, during the processing of the data it may be necessary to filter it again. These additional circuits are also provided as Appendix F.2.

D. MICROCOMPUTER INTERFACE

The flow chart, provided as Appendix F.3, illustrates the Z-80 assembly language program, Appendix F.4, that was needed to drive the A/D circuit and collect the speech data.

The program, A2D.ASM, was also useful in testing, step by step, the proper operation of the circuit.

The Z-80A micro-processor is at the heart of the system and the software designed to drive it is assembled using the Macro Assembler (M80) and linked to the Prolog station using Link software (L80). For more information on these procedures refer to the Bibliography.

1. Sampling Rate

The sampling rate is not arbitrary. It is a function of the software. In assembly language programming each step that the microprocessor goes through takes a specific amount of time. We will refer to a measure of time as a T state. Each T-state equals the inverse of the clock rate interfaced with the Z-80 chip. Since we are using a 4 MHz clock, one T-state equals 250 nano-seconds.

Every command line in the assembly program, including the command 'No Operation' or NOP, requires several T-states to accomplish its task. We are interested in the interval of time it takes from one sample to the next, and then we modify the software accordingly.

This program has a delay loop in it (labeled DELAY) to slow down the data acquisition to 10K samples/second. If it did not have the delay loop in it, it would easily sample at 23K samples/second. Since each utterance was limited to

less than one second, 10K samples is workable and does not present prohibitive record lengths.

E. DATA FILE SETUP AND MANIPULATION

Once the data is collected and stored in the Prolog's 32K buffer, it is uploaded onto a Zenith 100 formatted floppy disk and stored in an appropriately titled HEX file. A sample of a typical segment of data is provided as Figure 11.

```
:105AE000807E807D7E7E797776797E7E7D7F80833E5
:105AF00085858686868583807E7F80807C7B7D7E93
:105B0000808180808180817F807F807D7D7C7D7EA3
:105B10007E7C7E7E7E7F808080807E807F807D7D7D9F
:105B20007E7E7D7C7773777F83807E80878D8A877A
:105B3000868586837E7C7D7D7D7B797E809381806A
:105B400080818282807E7E7C7C7D7C7D7E7E80806D
```

Hexadecimal Data File Segment

Figure 11.

The file is in Intel-Hex format. The colon starts off each line. The following '10' tells us that the line is full of data. The next four digits indicate the memory location in the buffer. Every two bits following the memory location represents a byte of information.

Following a double 0, there are 16 records of data, and then a checksum byte at the very end. For our purposes the first nine digits and the last two digits are of no use. The Intel-Hex file is already in ASCII format.

An Osborne microcomputer was used to transfer data from the Zenith 100 formatted floppy disk to a Kaypro formatted

floppy disk. Since the data is needed in integer form to do the necessary processing, a program was written [App. F.5], in Microsoft Basic Language (MBASIC), to convert the data files from hexadecimal into the equivalent integer values.

Finally, the data is ready for processing. Since the software was already written on the IBM-3033 to process and display the data, it was sent there via a 1200 baud modem, and processed.

<u>APPENDIX F.2</u>

<u>CIRCUIT DIAGRAMS FOR THE DATA ACQUSITION SYSTEM</u>

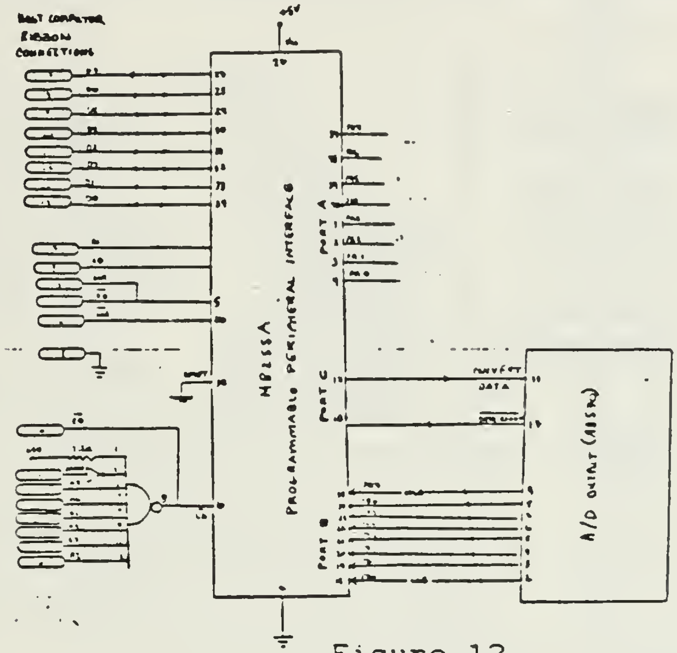
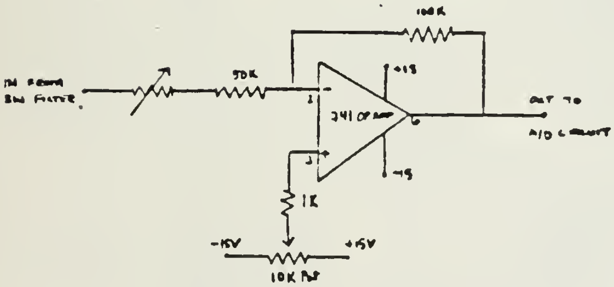


Figure 12.

Pin Out of the 8255 Programmable Peripheral Interface



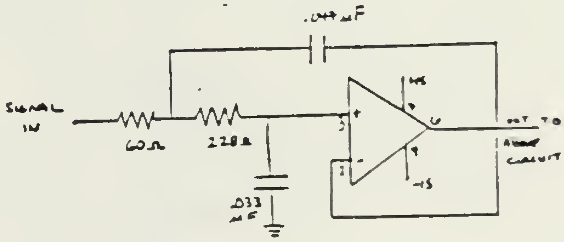
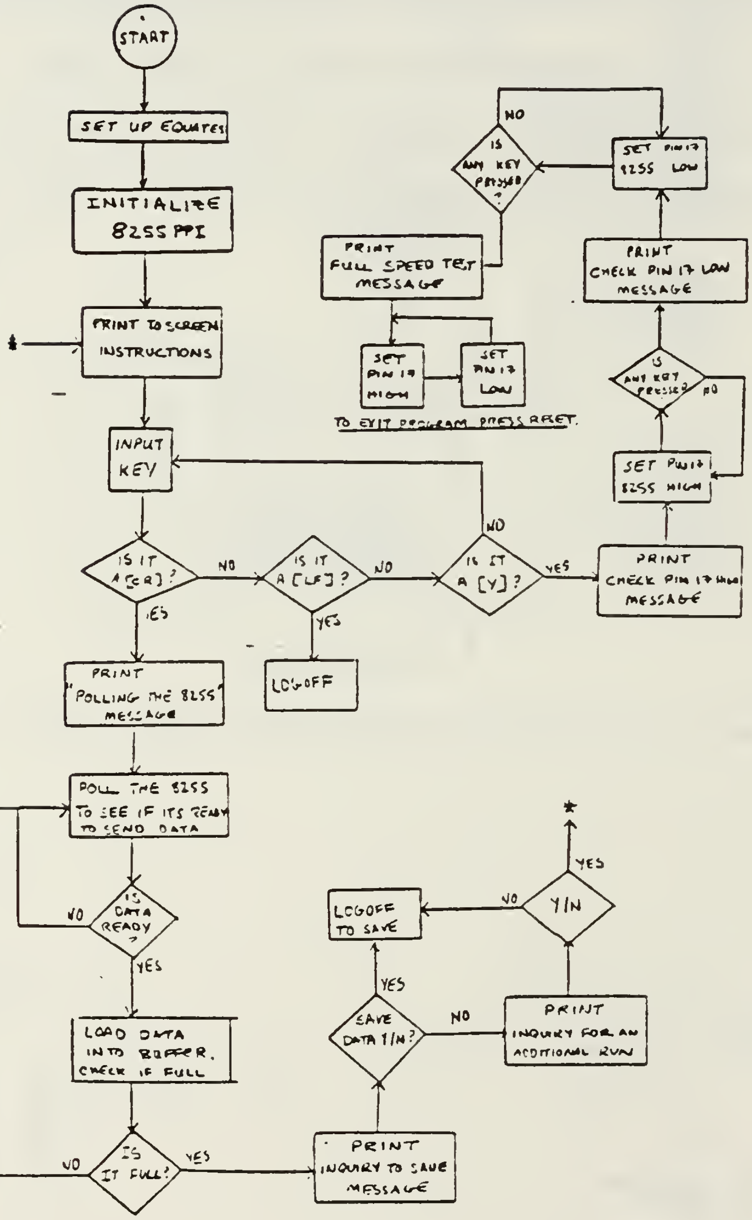


Figure 13.

Adjustable Gain and DC-Offset Circuit

Figure 14.

Low Pass, 2 Pole Butterworth Filter



SOFTWARE FLOW CHART FOR THE ASSEMBLY LANGUAGE DATA ACQUISITION PROGRAM

APPENDIX F.4 - ASSEMBLY LANGUAGE PROGRAM

This program is designed to acquire speech data.

```
;******************************************************************
;
;       THIS PROGRAM IS DESIGNED TO CAPTURE AND SAVE TO THE PROLOG'S MEMORY
;       OUTPUT 8-BIT DATA FROM AN EXTERNAL CIRCUIT.  TO TRANSFER THE PROLOG'S
;       MEMORY TO THE HOST COMPUTER, FOR STORAGE ON A DISK FILE, USE ZMDS.
;
;******************************************************************
ACQUIRE:                                  ;EQUATES
        PPI     EQU     0FCH              ;8255 PARALLEL PORT ADDRESS
        PORTA   EQU     PPI               ;PORT A ADDRESS
        PORTB   EQU     PPI+1             ;PORT B ADDRESS
        PORTC   EQU     PPI+2             ;PORT C ADDRESS
        PPICR   EQU     PPI+3             ;PPI CONTROL REGISTER
        MODEF   EQU     09AH              ;MODE 0, A IN, B IN, CU IN, CL OUT
        CWSET   EQU     007H              ;CONTROL WORD, SET BIT 3 HIGH
        CWRESET EQU     006H              ;CONTROL WORD, RESET BIT 3 LOW
        BDOS    EQU     005H              ;STANDARD CP/M COMMAND
        CR      EQU     00DH              ;CARRAGE RETURN ASCII CODE
        LF      EQU     00AH              ;LINE FEED ASCII CODE
        BEND    EQU     06EH              ;HIGH BYTE OF END OF MEMORY BUFFER
;CHANGE BEND TO 07FH WHEN THROUGH WITH PIZSIL & BUFF TO 5000H
;******************************************************************
;                       MAIN PROGRAM
;******************************************************************
        LD      SP,STACK        ;INITIALIZES THE STACK
MAIN:   CALL    INIT            ;INITIALIZES 8255 PARALLEL PORT INTERFACE
        LD      DE,HEAD         ;PRINTS INSTRUCTIONS TO THE SCREEN
        CALL    PRINT           ;PRINT SUBROUTINE
KEYIN1: CALL    GET             ;OBTAINS THE KEY BOARD ENTRY
        CP      LF              ;IS THE INPUT KEY A LINE FEED?
        JR      NZ,KEYIN2       ;IF IT IS NOT A ZERO, IS IT A [CR]?
        CALL    LOGOFF          ;SINCE IT IS A ZERO, RETURN TO THE OPSYSTM
KEYIN2: CP      00DH            ;IS IT A CARRAGE RETURN [CR]?
        JR      NZ,KEYIN3       ;IF IT IS NOT A [CR], LOOP, OTHERWISE CONTINUE
        JP      START           ;GO TO IT
KEYIN3: CP      059H            ;IS IT A Y FOR YES TEST IT?
```

```
        JP      NZ,KEYIN1       ;TRY ANOTHER INPUT
        LD      DE,TESTCR       ;TEST THE CIRCUIT MESSAGE.
        CALL    PRINT           ;PRINT THE MESSAGE
        CALL    TSTPTS          ;TEST 8255 PORTS SUBROUTINE
        JP      MAIN            ;START ALL OVER AGAIN
START:  LD      DE,POLLING      ;POLL 8255 MESSAGE
        CALL    PRINT           ;PRINT SUBROUTINE
        LD      HL,BUFF         ;STARTING ADDRESS OF MEMORY (JUST A COUNTER)
        LD      IX,BUFF         ;STARTING ADDRESS OF MEMORY ON INPUT BUFFER
        CALL    RXDATA          ;POLL 8255 SUBROUTINE, CHECK FOR DATA READY
        CALL    GET             ;KEY BOARD INQUIRY FROM POLL SUBROUTINE
        CP      059H            ;IS THE KEY BOARD ENTRY A 'Y' FOR YES?
        JR      NZ,MAIN         ;DONT WANT TO SAVE BUFFER?
        LD      DE,GOODTIME     ;GOOD TIME TO SAVE THE BUFFER TO DISK FILE
        CALL    PRINT           ;PRINT SUBROUTINE
        CALL    GET             ;KEY BOARD INQUIRY FROM DUMP SUBROUTINE
        CP      059H            ;IS THE RESPONSE YES?
        JR      NZ,MAIN         ;IF ANOTHER RUN IS DESIRED GO TO THE BEGINNING
        LD      DE,FINISH       ;FINAL COMMENTS PRIOR TO LOGOFF
        CALL    PRINT           ;PRINT SUBROUTINE
        CALL    LOGOFF          ;RETURN TO THE OPERATING SYSTEM
;
;*****************************************************************************
;                               SUBROUTINES
;*****************************************************************************
;
INIT:   LD      A,MODEF         ;8255 INITIALIZATION WORD
        OUT     (PPICR),A       ;MODE 0, A IN, B IN, CU IN, CL OUT
        RET                     ;RETURN TO THE MAIN PROGRAM
;
LOGOFF: JP      000H            ;JUMP TO THE OPERATING SYSTEM
;
RXDATA: LD      A,CWSET         ;DATA COVERT SIGNAL VIA 8255
        OUT     (PPICR),A       ;SEND IT OUT TO AD570 CHIP TO SAMPLE SIGNAL
        LD      A,CWRESET       ;END DATA CONVERT SIGNAL
        OUT     (PPICR),A       ;SEND IT VIA 8255 CONTROL REGISTER
```

```
POLL:   IN      A,(PORTC)       ;DATA IS READY IF BIT 4 OF PORT C IS HIGH
        AND     080H            ;IS READY HIGH?
        JP      NZ,POLL         ;LOOP UNTIL IT IS READY
        IN      A,(PORTB)       ;LOAD ACCUMULATOR W/CONTENTS OF PORT A
        LD      (IX),A          ;CONTENTS OF A BECOME CONTENTS AT IX ADDRESS
        INC     HL              ;INCREMENTS THE COUNTER (RR<--RR+1)P.265Z80BK
        INC     IX              ;INCREMENTS THE POINTER
        LD      A,H             ;QUERY THE BUFFER, IS IT FULL?(HIGH BIT ONLY)
        CP      BEND            ;END OF BUFFER COMPARRISON TEST
        JP      NZ,LOAD         ;BUFFER OKAY, IS THERE STILL DATA AVAILABLE?
        LD      DE,FULL         ;BUFFER FULL MESSAGE
        CALL    PRINT           ;PRINT SUBROUTINE
        RET                     ;RETURN TO THE MAIN PROGRAM
LOAD:   LD      B,014H          ;WILL DECREMENT THIS 11 TIMES FOR DELAY
DELAY:  DEC     B               ;B=B-1
        NOP                     ;NOP PROVIDES 4T STATES FREE
        LD      A,B             ;PREPARE TO COMPARE THE DECREMENT WITH 09H
        CP      09H             ;COMPAIRING A WITH 09H
        JP      NZ,DELAY        ;JUMP TO DELAY IF ITS NOT ZERO
        JP      RXDATA          ;LOOP UNTIL DATA READY COMES LOW
;
TSTPTS: LD      A,CWSET         ;START DATA CONVERT SIGNAL
        OUT     (PPICR),A       ;SEND IT TO THE AD570
        LD      DE,ONE          ;MESSAGE TO CHECK FOR A HIGH ON PIN 17(8255)
        CALL    PRINT           ;PINT MESSAGE SUBROUTINE
        CALL    GET             ;PRESS CR WHEN READY TO CONTINUE
        CP      CR              ;IS IT A CARRAGE RETURN
        JP      NZ,TSTPTS       ;IF NOT TRY THIS TEST AGAIN
LOW:    LD      A,CWRESET       ;STOP DATA CONVERT SIGNAL
        OUT     (PPICR),A       ;SEND IT TO THE AD570
        LD      DE,TWO          ;MESSAGE FOR SECOND TEST
        CALL    PRINT           ;IS PIN 17 NOW A LOW(ZERO ON 8255)?
        CALL    GET             ;GET A CR TO CONTINUE
        CP      CR              ;IS IT A CR?
        JP      NZ,LOW          ;IF NOT TRY THIS PORTION OF THE TEST AGAIN
        LD      DE,THREE        ;FULL SPEED TEST MESSAGE
        CALL    PRINT           ;MESSAGE SUBROUTINE
TEST:   LD      A,CWSET         ;START DATA CONVERT SIGNAL
```

```
        OUT     (PPICR),A          ;SEND IT TO THE AD570
        LD      A,CNRESET          ;STOP DATA CONVERT SIGNAL
        OUT     (PPICR),A          ;SEND IT TO THE AD570
WAIT:   IN      A,(PORTC)          ;CHECK FOR DATA READY
        AND     080H               ;1000 0000 BINARY
        JP      NZ,WAIT            ;WAIT IF ITS NOT READY
        JP      TEST               ;OTHERWISE KEEP THE TEST GOING
        RET                        ;RETURN TO THE MAIN PROGRAM
;
;***************************************************************************
;                 SUBROUTINES INVOLVING CP/M BDOS CALLS
;***************************************************************************
;
GET:                       ;GET 1 CHARACTER FROM THE KEYBOARD
        LD      C,1        ;BDOS FUNCTION #1
        CALL    BDOS       ;STANDARD CP/M BDOS CALL
        RET                ;RETURN TO MAIN PROGRAM
;
PRINT:                     ;PRINT A MESSAGE
        LD      C,9        ;BDOS FUNCTION #9
        CALL    BDOS       ;STANDARD CP/M BDOS CALL
        RET                ;RETURN TO MAIN PROGRAM
;
;***************************************************************************
;                              MESSAGES
;***************************************************************************
;
HEAD:   DB      'THIS PROGRAM IS DESIGNED TO BE MENU DRIVEN. THE',CR,LF
        DB      'SOFTWARE SAMPLES AN INPUT FROM AN A/D CIRCUIT AND',CR,LF
        DB      'STORES DATA IN THE PROLOGS 32K BUFFER. IT THEN',CR,LF
        DB      'GIVES YOU TIME TO STORE THE DATA TO YOUR DISK FOR',CR,LF
        DB      'FUTURE PROCESSING USING ZMDS ON THE HOST COMPUTER.',CR,LF
        DB      'TO SAMPLE DATA, PRESS [CR]. TO TERMINATE THIS ',CR,LF
        DB      'PROGRAM PRESS THE LINE FEED. TEST,PRESS Y.',CR,LF
        DB      CR,CR,LF,'$'
;
```

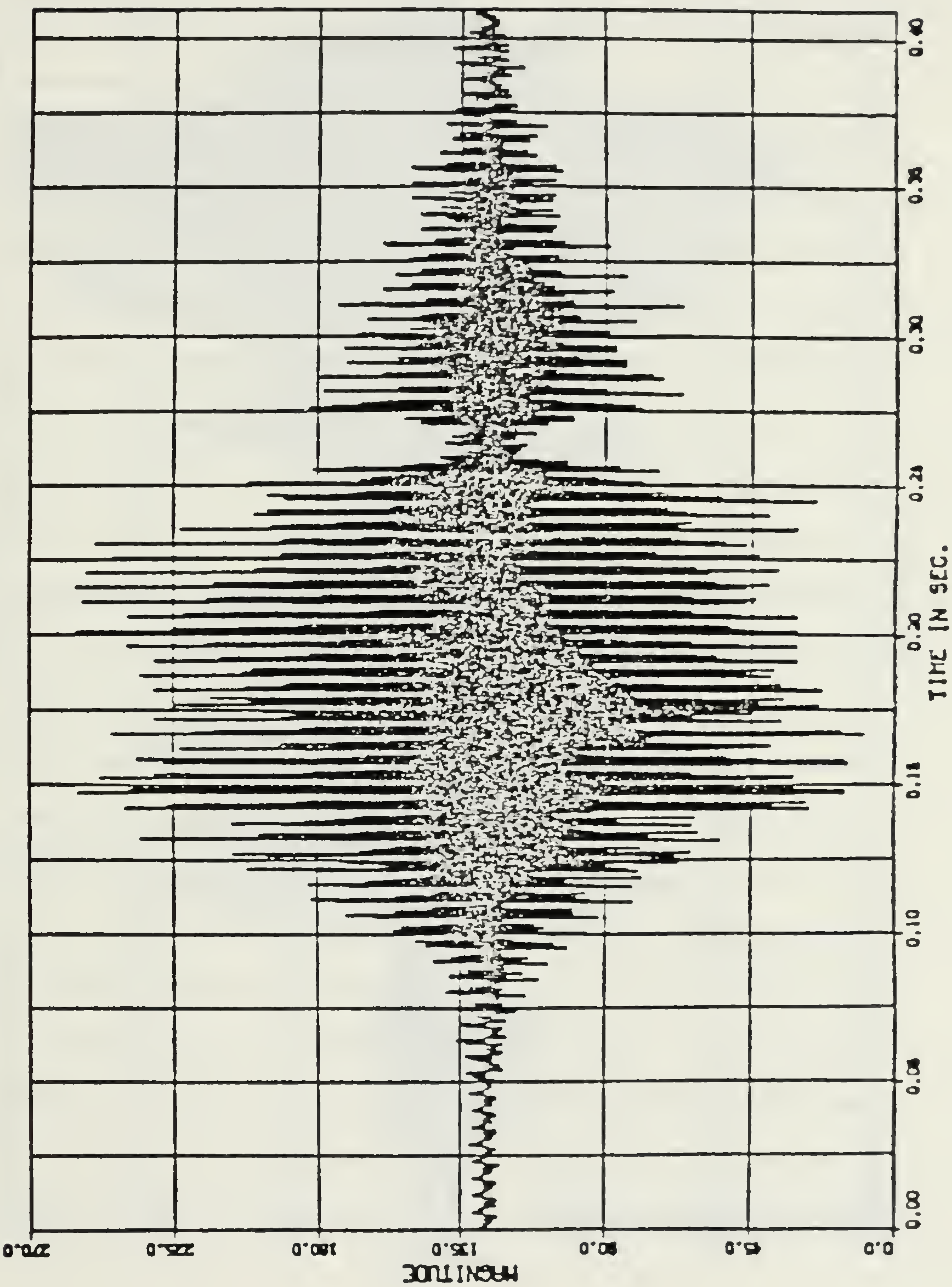
```
POLLING:          DB        'I AM CURRENTLY POLLING THE 8255 FOR A ',CR,LF
                  DB        'DATA READY SIGNAL. AT WHICH TIME I WILL ',CR,LF
                  DB        'COLLECT DATA TO THE BUFFER.',CR,LF
                  DB        'THE EXTERNAL CIRCUIT WILL INITIATE',CR,LF
                  DB        'RECEIVING DATA TO THE BUFFER!',CR,LF,LF,'$'
;
FULL:   DB        'THE BUFFER IS FULL!! DO YOU WANT TO SAVE ITS ',CR,LF
        DB        'CONTENTS TO YOUR DISK? (Y/N)',CR,LF,LF,'$'
;
GOODTIME:         DB        'THIS IS A GOOD TIME TO SAVE YOUR DATA TO ',CR,LF
                  DB        'YOUR DISK. TO DO THAT LOGOFF BY PRESSING',CR,LF
                  DB        'ANY KEY, OR IF YOU WANT ANOTHER RUN,TYPE',CR,LF
                  DB        'A Y FOR YES, LETS DO IT AGAIN!  ',CR,LF,LF,'$'
;
FINISH: DB        'HOPE YOU GOT WHAT YOU NEEDED! GOOD LUCK!',CR,LF,LF,'$'
;
TESTCR: DB        'THIS IS TESTING THE CIRCUIT FOR WIRING AND',CR,LF
        DB        'AND LOGIC ERRORS. ',CR,LF,LF,'$'
;
ONE:    DB        'CHECK FOR A HIGH ON THE 8255, PIN 17.',CR,LF
        DB        'TO CONTINUE, PRESS [CR].',CR,LF,LF,'$'
;
TWO:    DB        'CHECK FOR A LOW ON THE 8255, PIN 17.',CR,LF
        DB        'TO CONTINUE, PRESS [CR].',CR,LF,LF,'$'
;
THREE:  DB        'FULL SPEED TEST,PRESS THE RESET BUTTON ',CR,LF
        DB        'ON THE PROLOG, AND G 4000 AGAIN. ',CR,LF
        DB        CR,LF,LF,'$'
;
BUFF:    DS       2800H               ;START ADDRESS OF MEMORY BUFFER IN PROLOG
;
         DS       020H                ;SPACE FOR 16 STACK LOCATIONS
STACK:   DS       1                   ;TOP OF STACK
;
         END      ACQUIRE
;
```

APPENDIX F.5-INTELHEX TO DECIMAL DATA CONVERSION PROGRAM

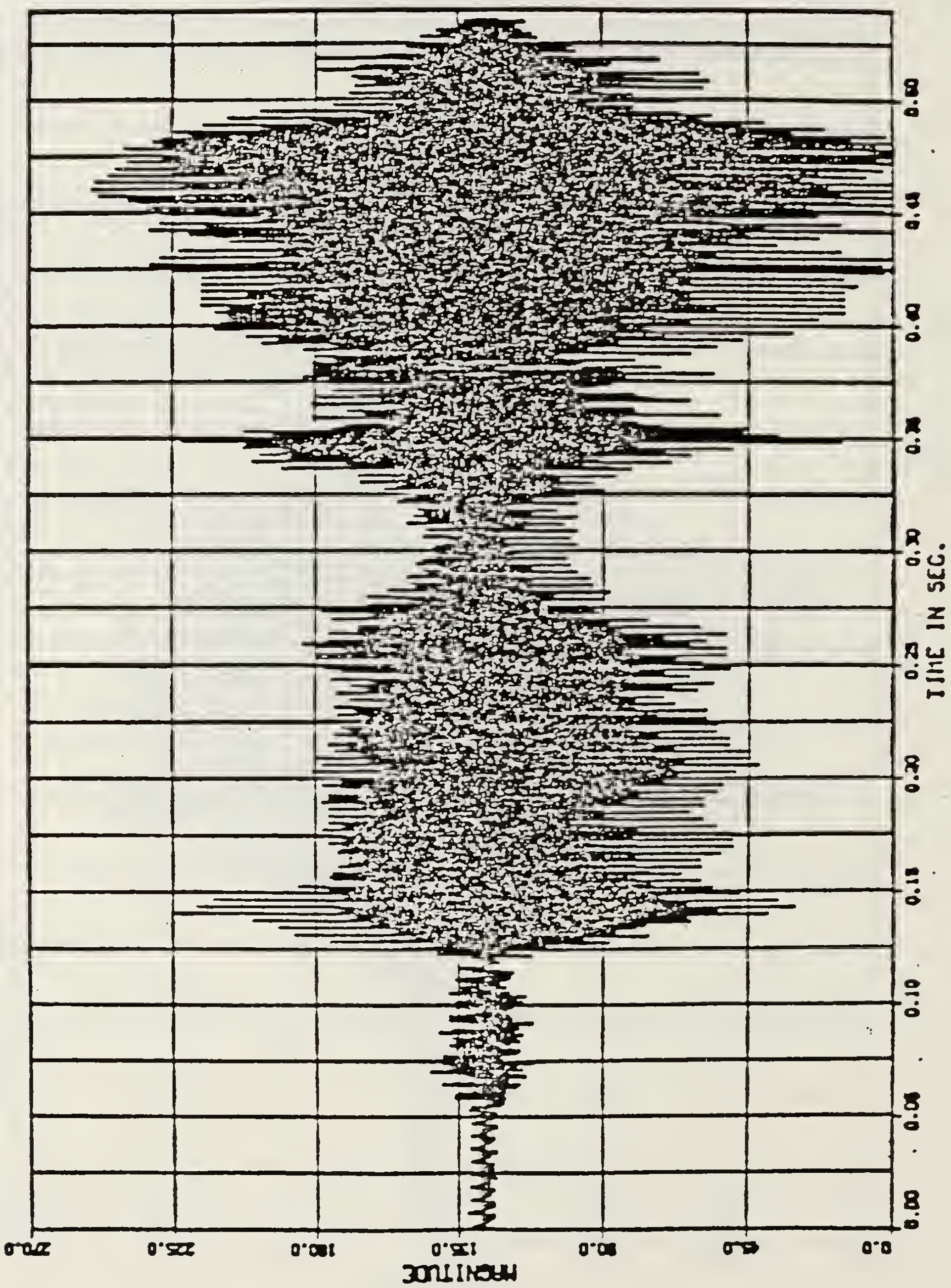
This program is designed to read a data file that is in Intelhex format and convert it to an integer file.

```
2  PRINT "INPUT FILE ":INPUT FI$
4  PRINT "OUTPUT FILE ":INPUT FO$
20 OPEN "O",2,FO$
30 OPEN "I",1,FI$
40 INPUT #1,IN$
60 IF MID$(IN$,2,1)="0" THEN CLOSE:GOTO 140
70 FOR I=10 TO 40 STEP 2
80 HX$=MID$(IN$,I,2)
90 V%=VAL("&H"+HX$)
95 IF I=40 THEN PRINT #2,USING"###";V% ELSE PRINT #2,USING"###";V%;
98 IF I=40 THEN PRINT USING "###";V% ELSE PRINT USING "###";V%;
100 NEXT I
125 PRINT
130 GOTO 40
140 PRINT "DONE" + CHR$(7)
```

APPENDIX G - READY-E

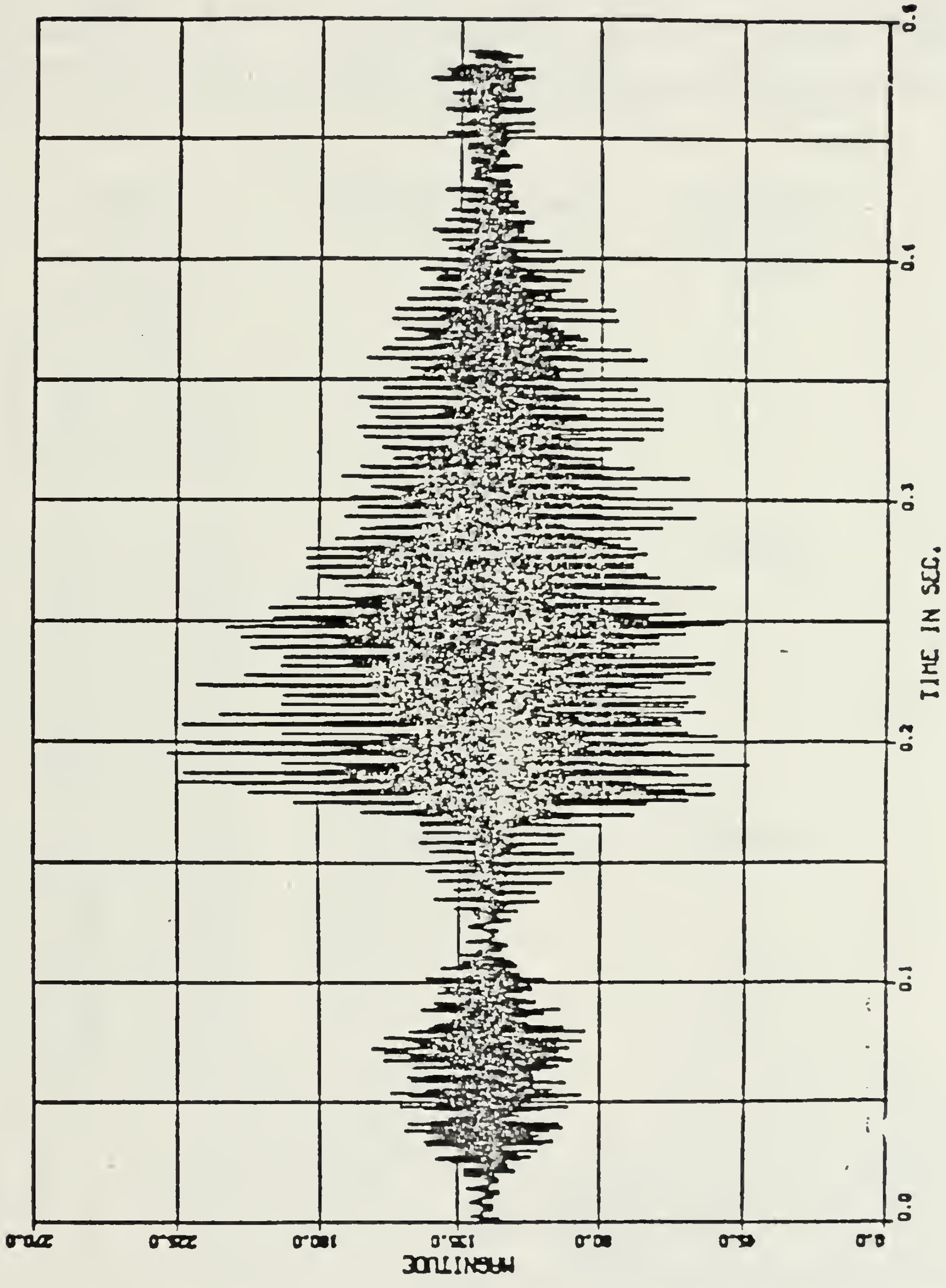


This is an example of the sampled utterence 'Ready'.



This is an example of the sampled utterence 'So What'.

APPENDIX I - SNEEZE-F



This is an example of the utterence 'Sneeze'.

APPENDIX J - SPEECH RESOLUTION SUMMARY

This table lists the actual ranges used by each utterence out of a possible 256 levels (from 0 to 255).

UTTERENCE	SCALE REFERENCE	RANGE
READY	MIDDLE-C	60 - 220
	D	52 - 230
	E	10 - 255
	F	0 - 255
	G	35 - 255
	A	10 - 220
	B	25 - 250
	HIGH-C	10 - 255
SO WHAT	MIDDLE-C	60 - 175
	D	10 - 220
	E	15 - 225
	F	10 - 210
	G	0 - 255
	A	8 - 202
	B	0 - 255
	HIGH-C	0 - 255
SNEEZE	MIDDLE-C	65 - 180
	D	48 - 255
	E	30 - 255
	F	45 - 255
	G	45 - 210
	A	52 - 220
	B	30 - 230
	HIGH-C	45 - 225

APPENDIX_K - SPEECH_REFLECTION_COEFFICIENT_PROGRAM

This program is designed to determine the reflection coefficients for a 12th order lattice filter model of speech. It yields a new set of K's every 10 ms.

```
C*****************************************************************
C
C       THIS PROGRAM IS DESIGNED TO EVALUATE THE FREQUENCY
C
C       CONTENT OR PATTERNS EXISTING  IN THE REFLECTION
C
C       COEFFICIENTS FOR EACH UTTERENCE.
C
C*****************************************************************
C
      REAL K(6,500),M(6,500),X(500),P,KX,
     *K1(100),K2(100),K3(100),K4(100),K5(100),K6(100)
      INTEGER I,N,J,IWK(7),M
      COMPLEX A(500)
      CALL TEK618
      CALL XNAME(' FREQUENCY (HZ) $',128)
      CALL YNAME(' MAGNITUDE $',128)
      CALL AREA2D(8.,10.)
      CALL HEADIN(' FREQUENCY CONTENT OF K(N) $',128,2.,1)
      CALL GRAF(0.,'SCALE',100.0,0.,'SCALE',50.0)
      M=6
      DO 10 J=1,40
      READ (2,100)P,K(1,J),K(2,J),K(3,J),K(4,J),K(5,J),K(6,J)
10    CONTINUE
      DO 30 J=41,64
      DO 20 I=1,6
      K(I,J)=0
20    CONTINUE
30    CONTINUE
      DO 50 I=1,6
      DO 40 J=1,64
      KX=FLOAT(K(I,J))
      A(J)=CMPLX(KX,0.0)
40    CONTINUE
      CALL FFT2C(A,M,IWK)
      DO 45 J=1,32
      M(I,J)=CABS(A(J))
50    CONTINUE
      DO 60 J=1,32
      K1(J)=M(1,J)
      K2(J)=M(2,J)
      K3(J)=M(3,J)
      K4(J)=M(4,J)
      K5(J)=M(5,J)
      K6(J)=M(6,J)
60    CONTINUE
      DO 70 I=1,32
      X(I)=I
```

```
70    CONTINUE
      CALL  CURVE(X,K1,32,0)
      CALL  ENDPL(0)
      CALL  CURVE(X,K2,32,0)
      CALL  ENDPL(1)
      CALL  CURVE(X,K3,32,0)
      CALL  ENDPL(2)
      CALL  CURVE(X,K4,32,0)
      CALL  ENDPL(3)
      CALL  CURVE(X,K5,32,0)
      CALL  ENDPL(4)
      CALL  CURVE(X,K6,32,0)
      CALL  ENDPL(5)
      CALL  DONEPL
      RETURN
      END
```

APPENDIX L - REFLECTION COEFFICIENT PATTERNS FOR SPEECH WAVE FORMS

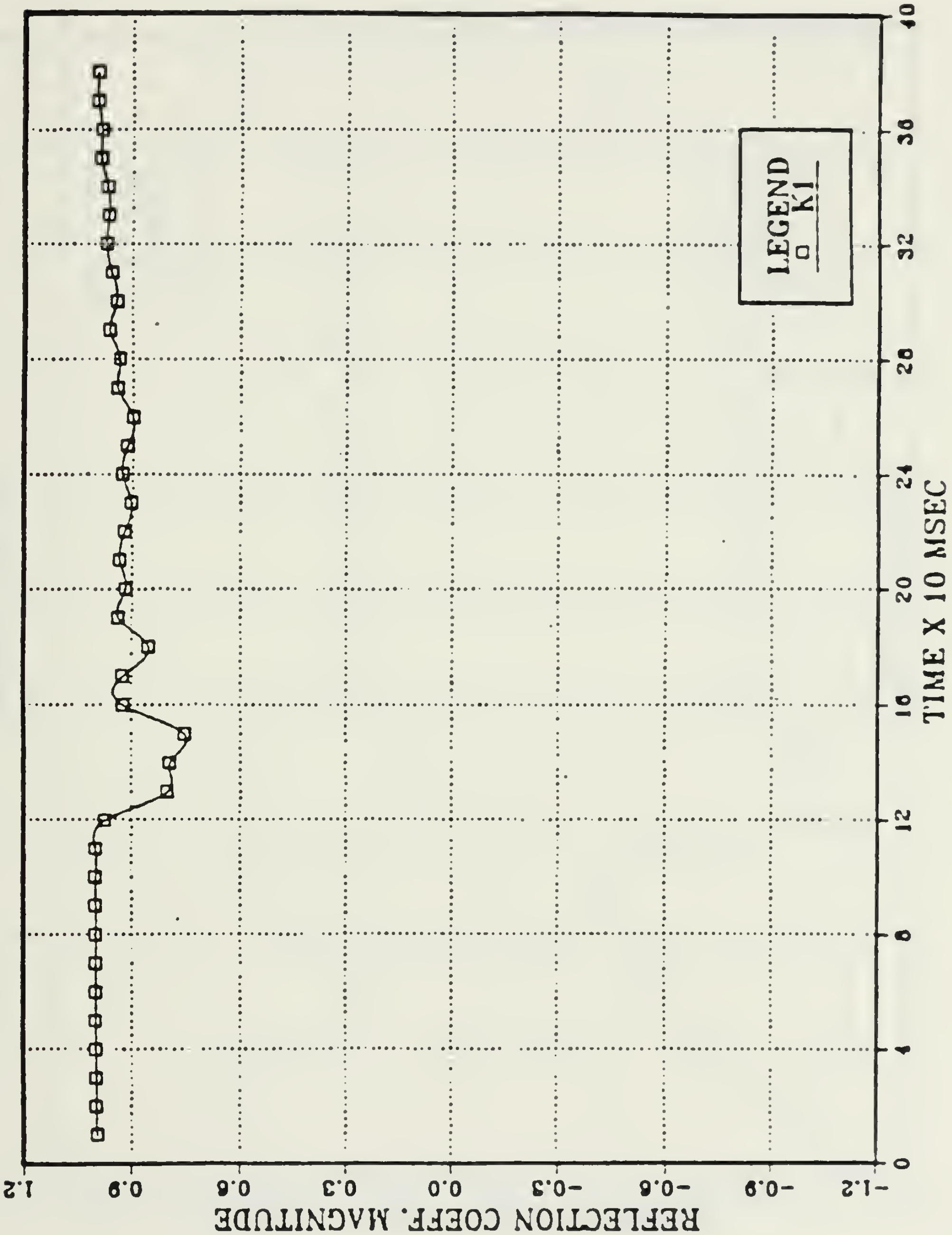


Figure L.1. Sneeze-E Pattern of Reflection Coefficient K1.

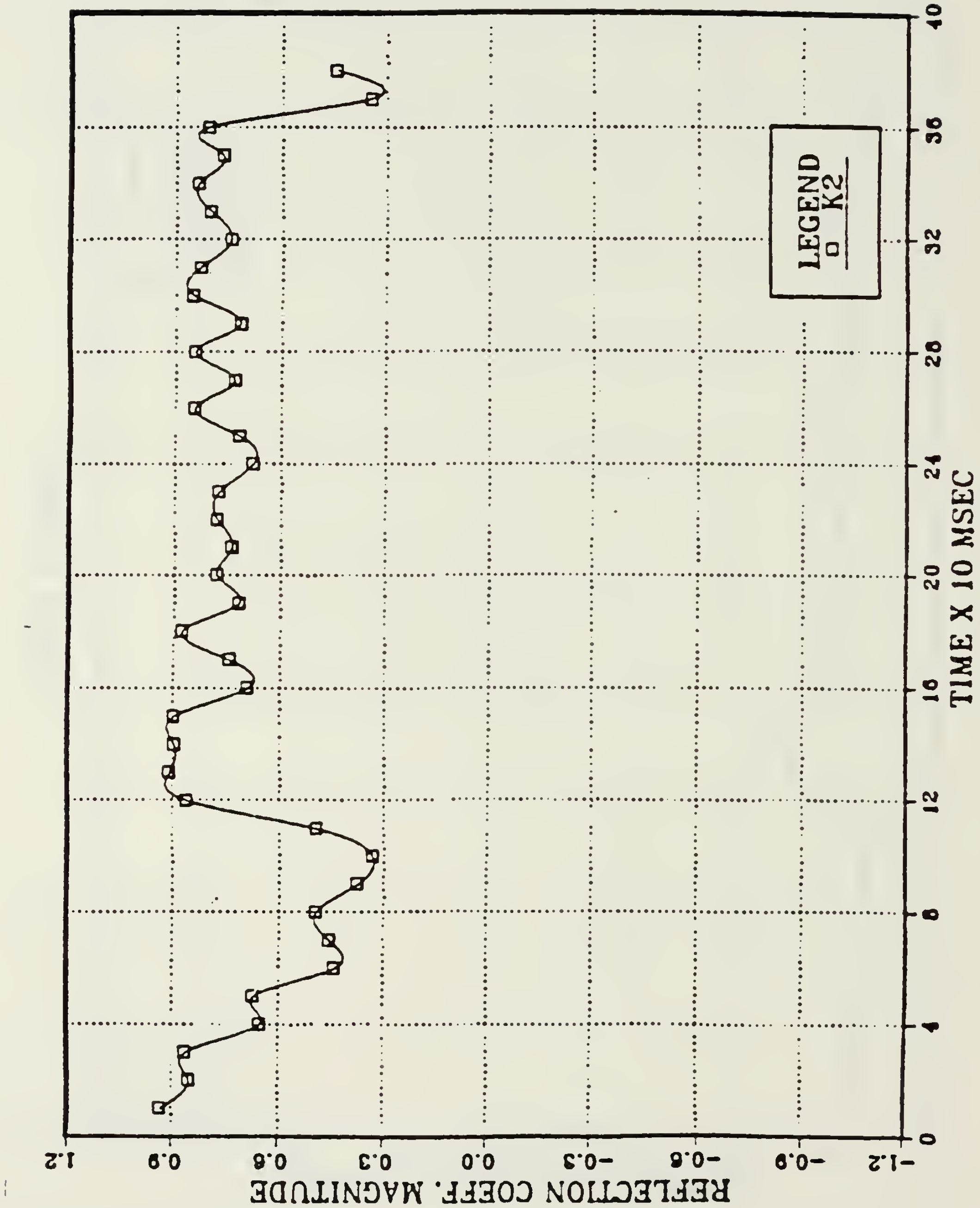


Figure L.2. Sneeze-E Pattern of Reflection Coefficient K2.

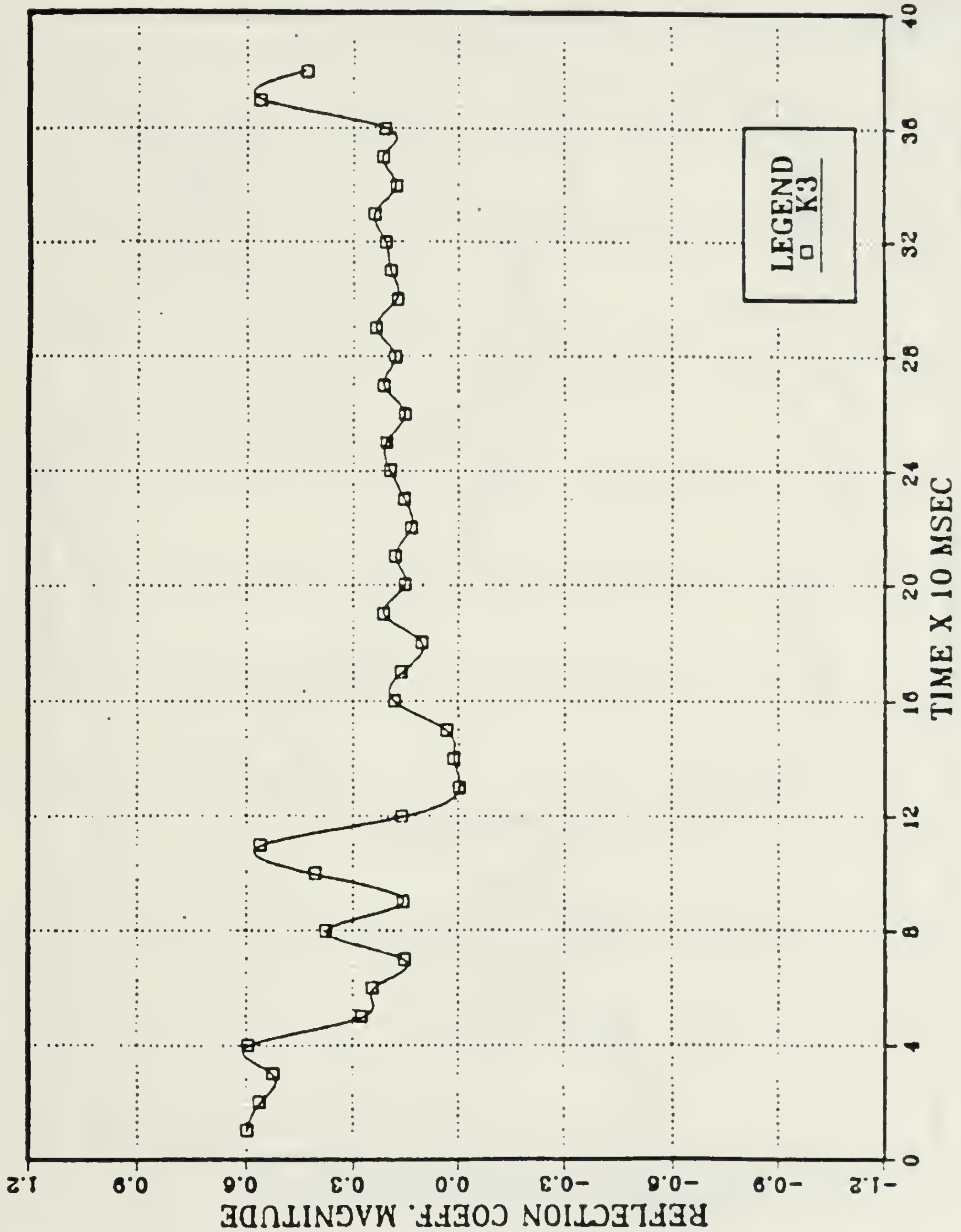


Figure L.3. Sneeze-E Pattern of Reflection Coefficient K3.

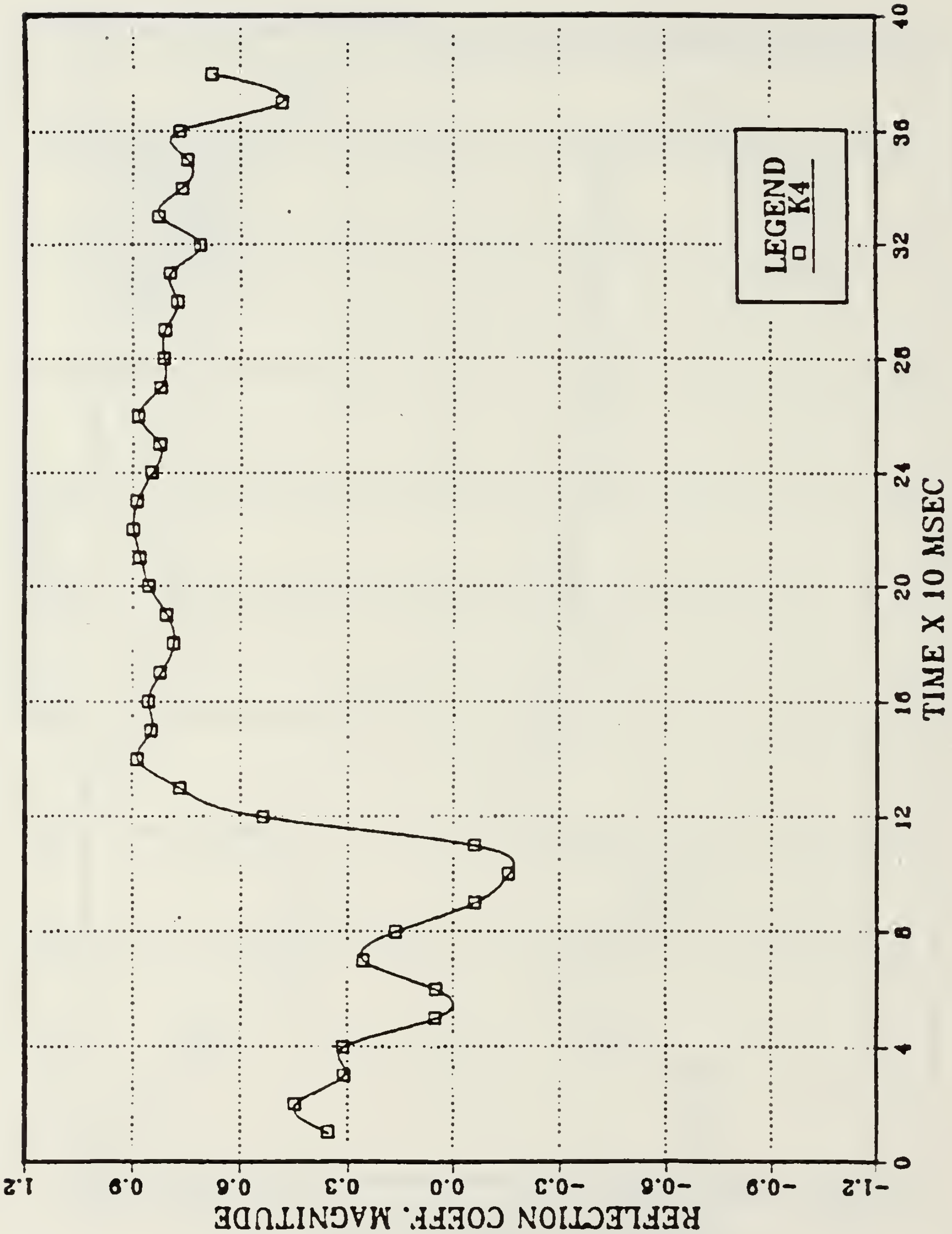


Figure L.4. Sneeze-E Pattern of Reflection Coefficient K4.

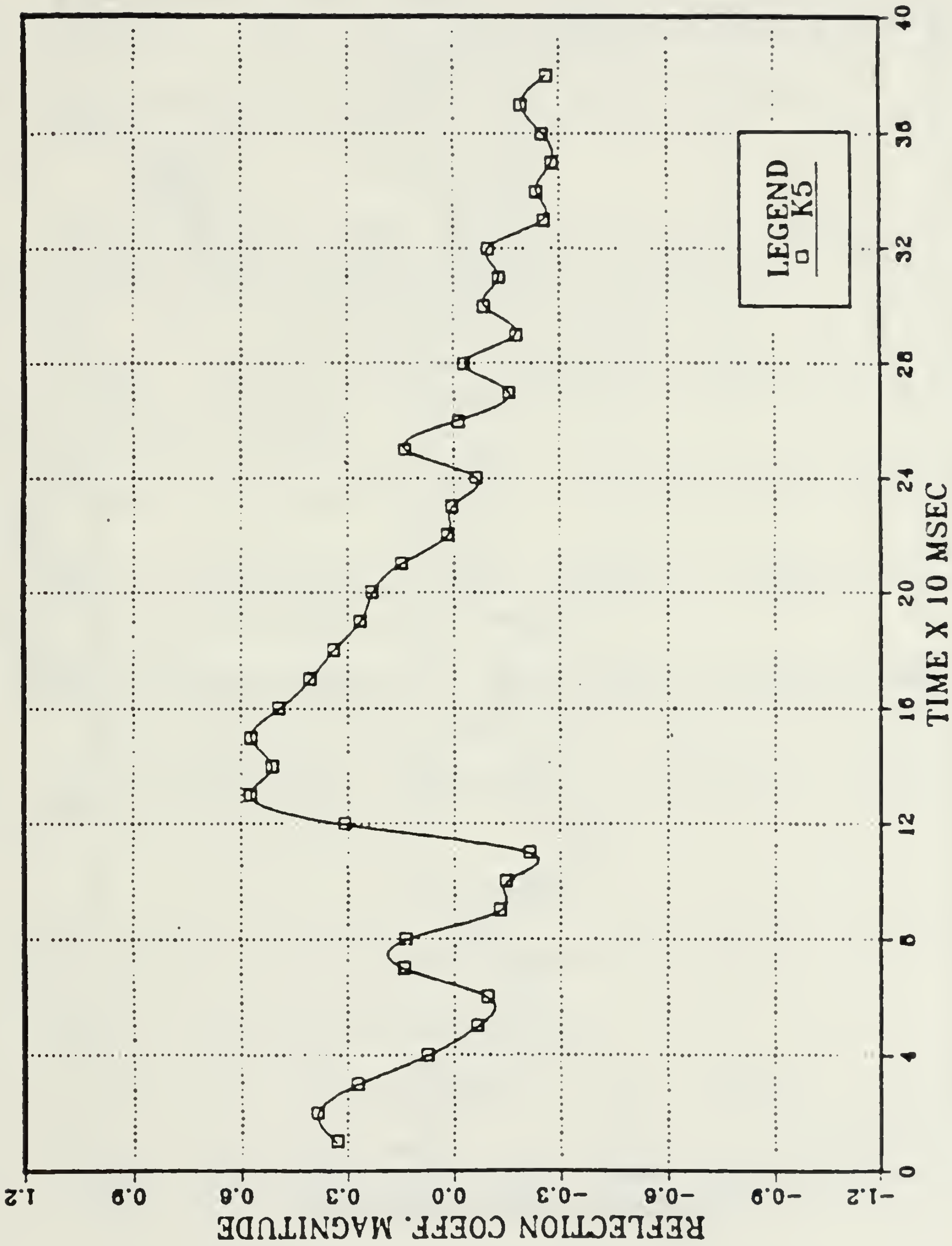


Figure L.5. Sneeze-E Pattern of Reflection Coefficient K5.

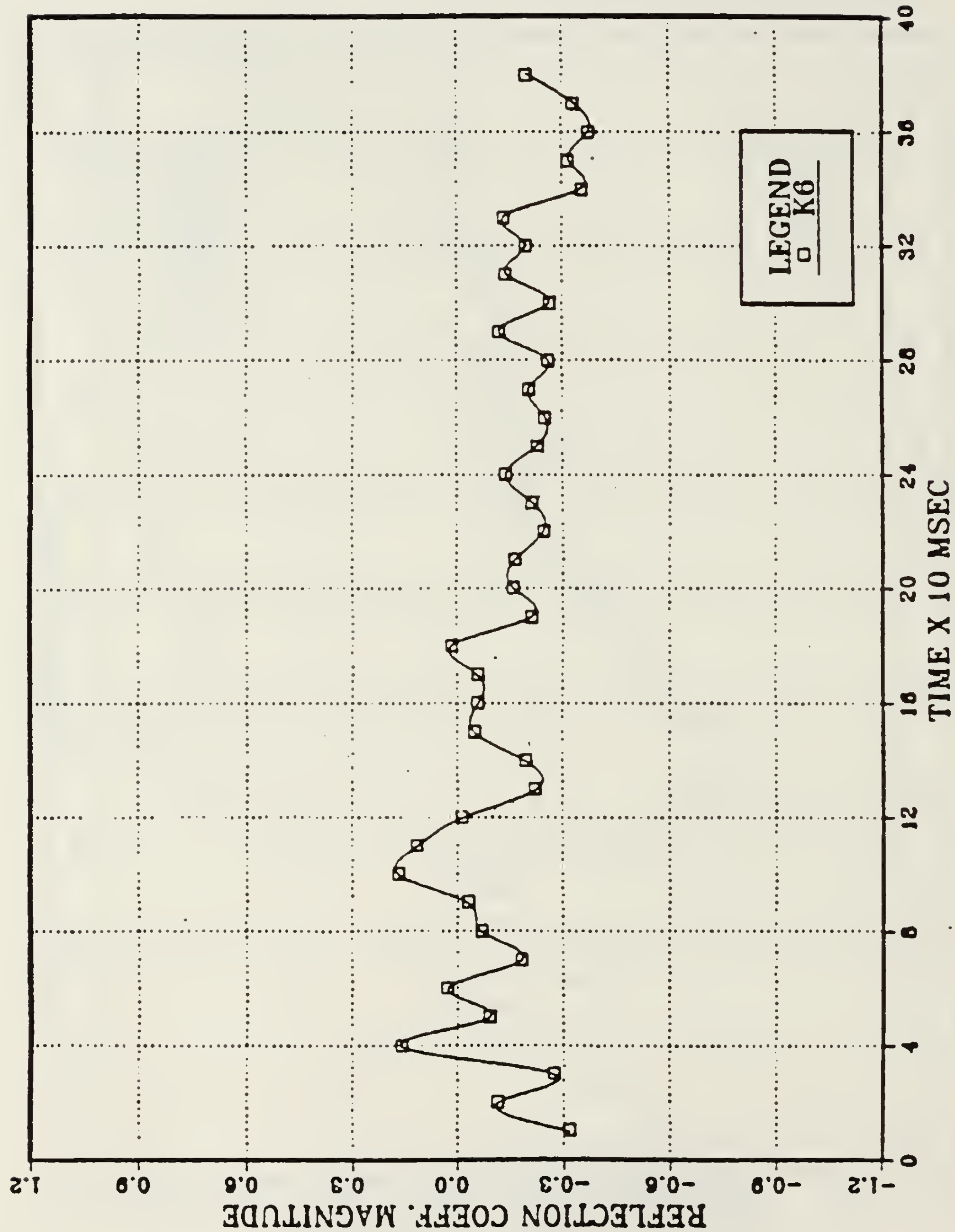


Figure L.6. Sneeze-E Pattern of Reflection Coefficient K6.

APPENDIX M - TREND ANALYSIS RESULTS

The following lists are the observations made on the reflection coefficient curves for each utterence.

"READY"

K1 - All pitches have relatively flat curves. The magnitudes vary slightly between +.8 and +1.0. The higher the pitch, the more defined the troughs are.

K2 - These curves all had the unique feature of sloping upward. They generally ranged from -.4 to +.9. No other correlation was noted.

K3 - A negative sloping tendency characterized this set of curves.

K4 - Each of these curves had a plateau. Ready B, however did not fit in with this set at all.

K5 - These curves seemed to stay within a similar range, .3 to -.7. Also several prominent peaks were uncorrelated.

K6 - No correlations were noted, however Ready-B was drastically different.

"SNEEZE"

K1 - Relatively flat curves. Ranges from .8 to 1.0.

K2 - Highly uncorrelated curves.

K3 - Also highly uncorrelated curves, however, more flat than K2.

K4 - MC and D are similarly flat, the rest seem correlated with a valley to an elevated flat plateau.

K5 - There seems to be a peak, then a declining trend in most of these curves. Again MC and D don't fit this observation and are generally flat.

K6 - There are several peaks, then relatively flat curves.

"SO WHAT"

K1 - Similarly flat patterns.

K2 - Highly uncorrelated with no recognizable patterns.

K3 - There is a prominent valley in all of the observations except A.

K4 - Highly uncorrelated with no recognizable patterns.

K5 - Highly uncorrelated with no recognizable patterns.

K6 - Highly uncorrelated with no recognizable patterns.

APPENDIX N - FAST FOURIER TRANSFORM PROGRAM

This program determines if there are any discrete frequencies existing within the reflection coefficient patterns.

```
FILE: ANAL      FORTRAN  A1

C*****************************************************************
C
C       DETERMINES 12 REFLECTION COEFFICIENTS FOR AN UTTERENCE OF
C
C       SPEECH AT DIFFERENT PITCH LEVELS DEPENDING ON THE DATA.
C
C                       READY. SO WHAT. SNEEZE
C
C*****************************************************************
C       OS - ORIGINAL SIGNAL
C       B - BLOCK OR SEGMENT OF DATA (100 PTS). T=10.0E-3
C       K(N) - REFLECTION COEFFICIENTS
C       AC - AUTOCORRELATION MATRIX (1XN)
C       FE - FORWARD ERROR. BE - BACKWARD ERROR
C       SUMN - E<OS(N)*OS(N+1):NUMERATOR
C       SUMD - (E(OS(N))**2):DENOMINATOR
C*****************************************************************
      INTEGER N.I.J.L.M.B.R.OS(5000)
      REAL SUMN.SUMD.AC(12).K(12).FE(12.5000).
     *BE(12.5000).MS(12)
      SUMN=0.0
      SUMD=0.0
      DO 5 J=1.4064.16
      READ(2.100)OS(J).OS(J+1).OS(J+2).OS(J+3).OS(J+4).OS(J+5).
     *OS(J+6).OS(J+7).OS(J+8).OS(J+9).OS(J+10).OS(J+11).OS(J+12).
     *OS(J+13).OS(J+14).OS(J+15)
5     CONTINUE
      DO 10 B=1.40
      DO 20 R=1.100
      N=R+(100*(B-1))
      SUMN=SUMN+(OS(N)*OS(N+1))
      SUMD=SUMD+((OS(N))**2)
20    CONTINUE
      K(1)=SUMN/SUMD
      FE(1.1)=0.0
      BE(1.1)=0.0
      DO 30 R=2.101
      J=R+(100*(B-1))
      FE(1.J)=OS(J)-(K(1)*OS(J-1))
      BE(1.J)=OS(J-1)-(K(1)*OS(J))
30    CONTINUE
      DO 40 N=2.12
      FE(N.1)=0.0
      BE(N.1)=0.0
      AC(N-1)=0.0
```

```
      MS(N-1)=0.0
      DO 50 R=2.101
      J=R+(100*(B-1))
      AC(N-1)=AC(N-1)+(FE(N-1.J)*BE(N-1.J-1))
      MS(N-1)=MS(N-1)+(FE(N-1.J)**2)
50    CONTINUE
      K(N)=AC(N-1)/MS(N-1)
      DO 60 R=2.101
      J=R+(100*(B-1))
      FE(N.J)=FE(N-1.J)-(K(N)*BE(N-1.J-1))
      BE(N.J)=BE(N-1.J-1)-(K(N)*FE(N-1.J))
60    CONTINUE
40    CONTINUE
C*****************************************************************
      WRITE(4.300)B.K(1).K(2).K(3).K(4).K(5).K(6)
      WRITE(9.300)B.K(7).K(8).K(9).K(10),K(11).K(12)
C*****************************************************************
      SUMN=0.0
      SUMD=0.0
10    CONTINUE
100   FORMAT(I3.I3.I3.I3.I3.I3.I3.I3.I3.I3.I3.I3.I3.I3.I3.I3)
300   FORMAT(I2.2X.F8.6.2X.F8.6.2X.F8.6.2X.F8.6.2X.F8.6.2X.F8.6)
      STOP
      END
```

APENDIX_O - FREQUENCY_CONTENT_OF_K(N)

This is an example of the output from the FFT program to determine if there are any discrete frequencies present in the reflection coefficient patterns.

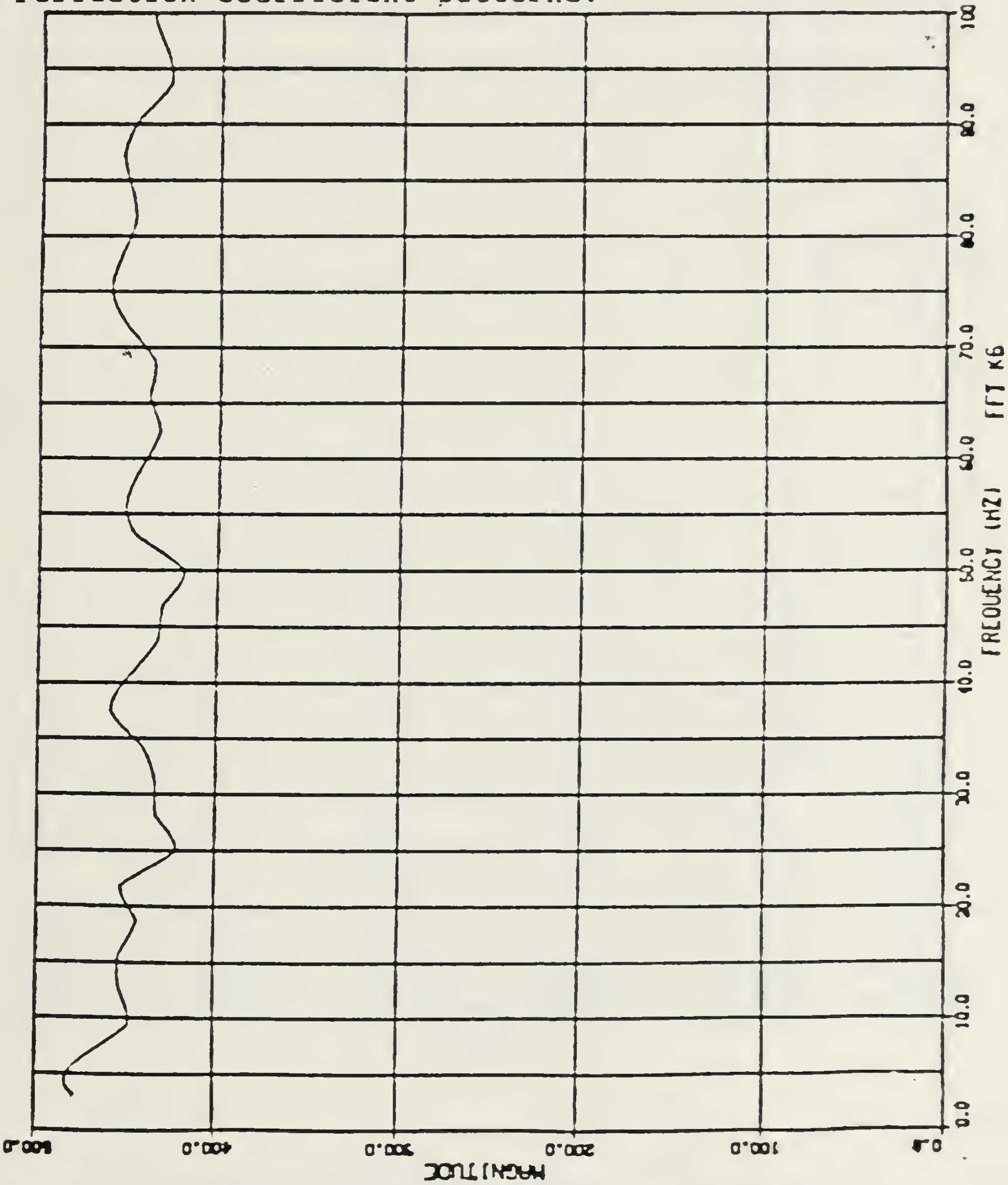


Figure O.1. Reflection Coefficient K6 for Utterence 'Ready-MC'.

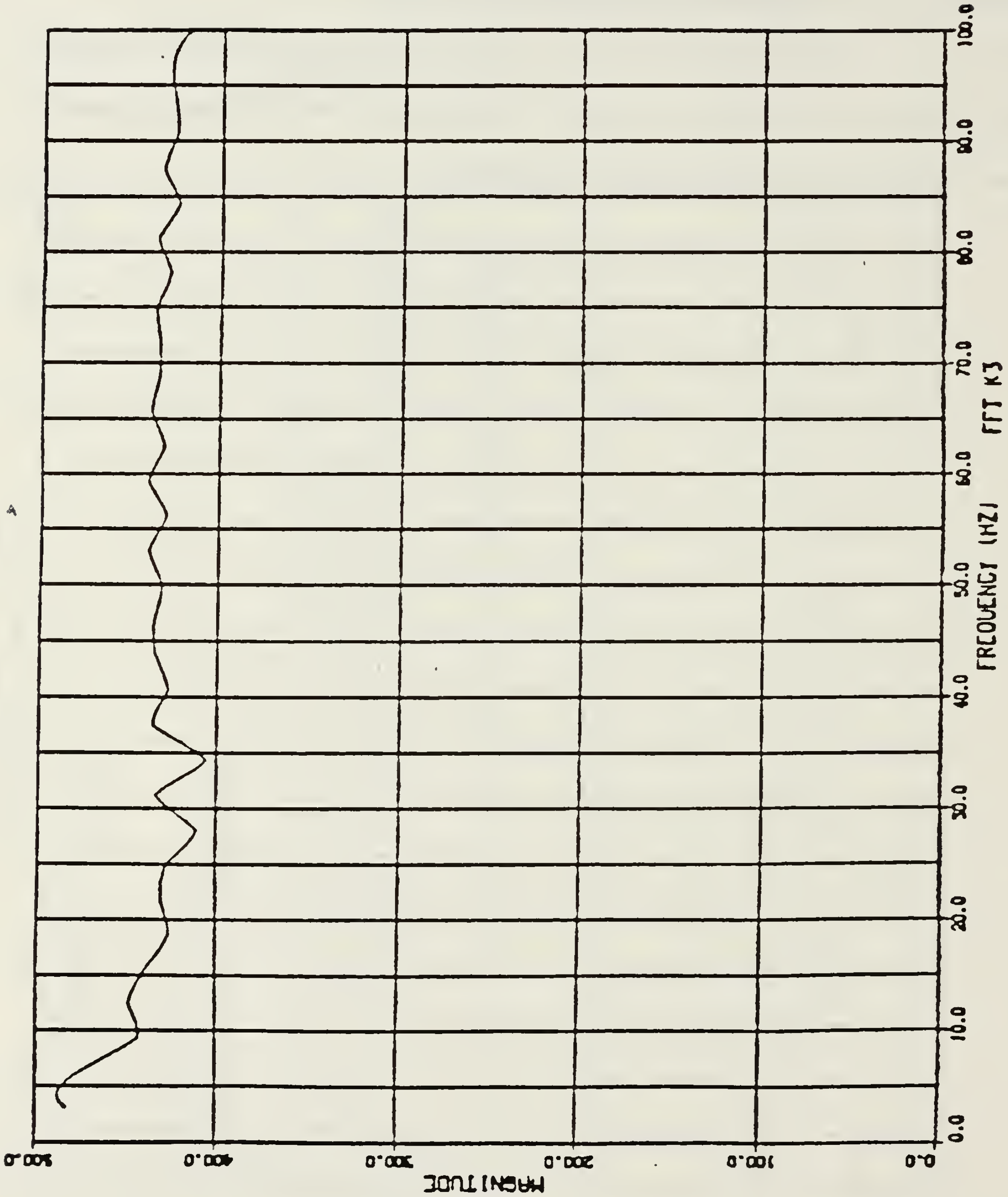


Figure 0.2. Reflection Coefficient K3 for
Utterence 'Ready-MC'.

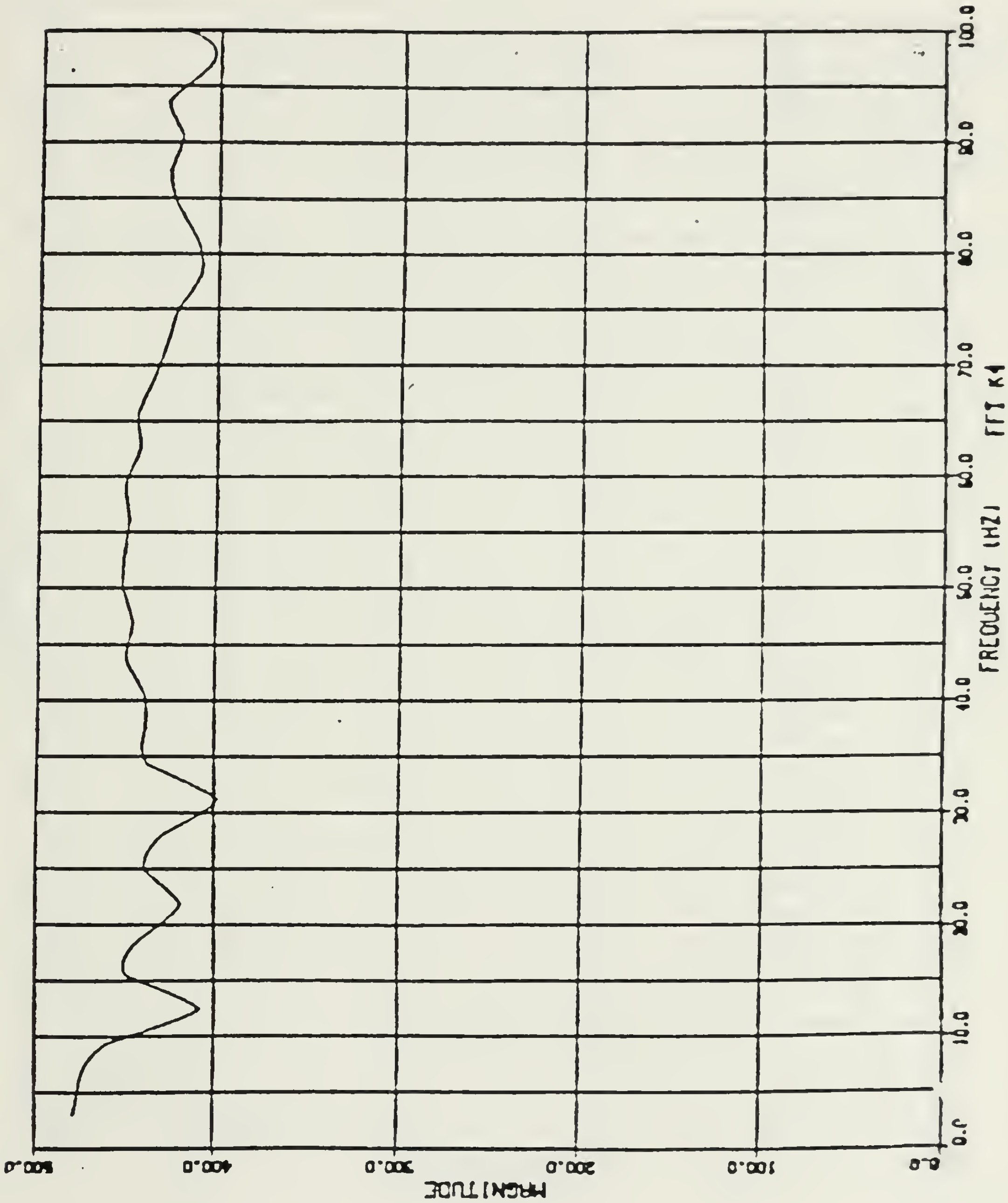


Figure O.3. Reflection Coefficient K4 for
Utterence 'Sneeze-MC'.

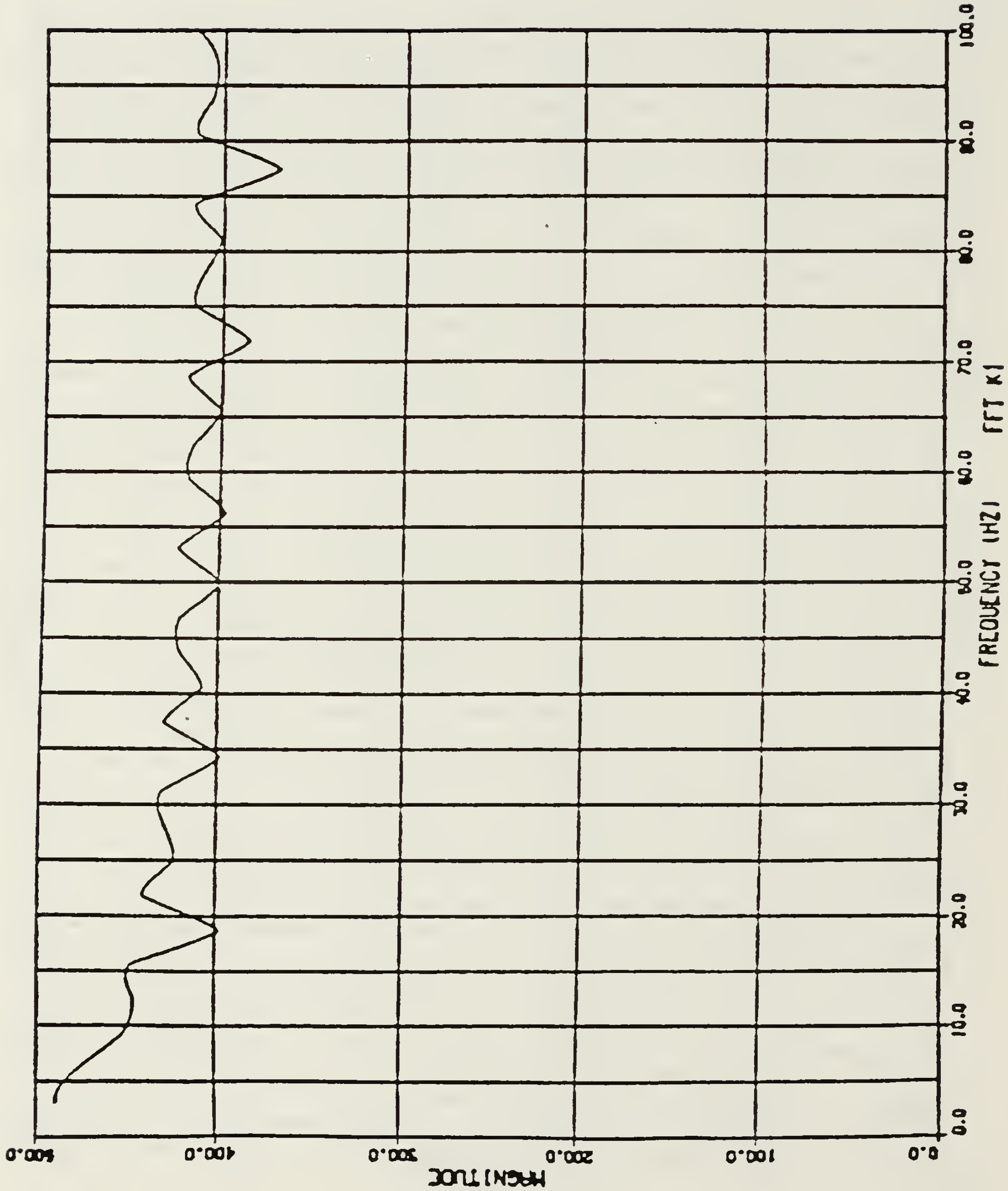


Figure O.4. Reflection Coefficient K1 for Utterence 'Sneeze-MC'.

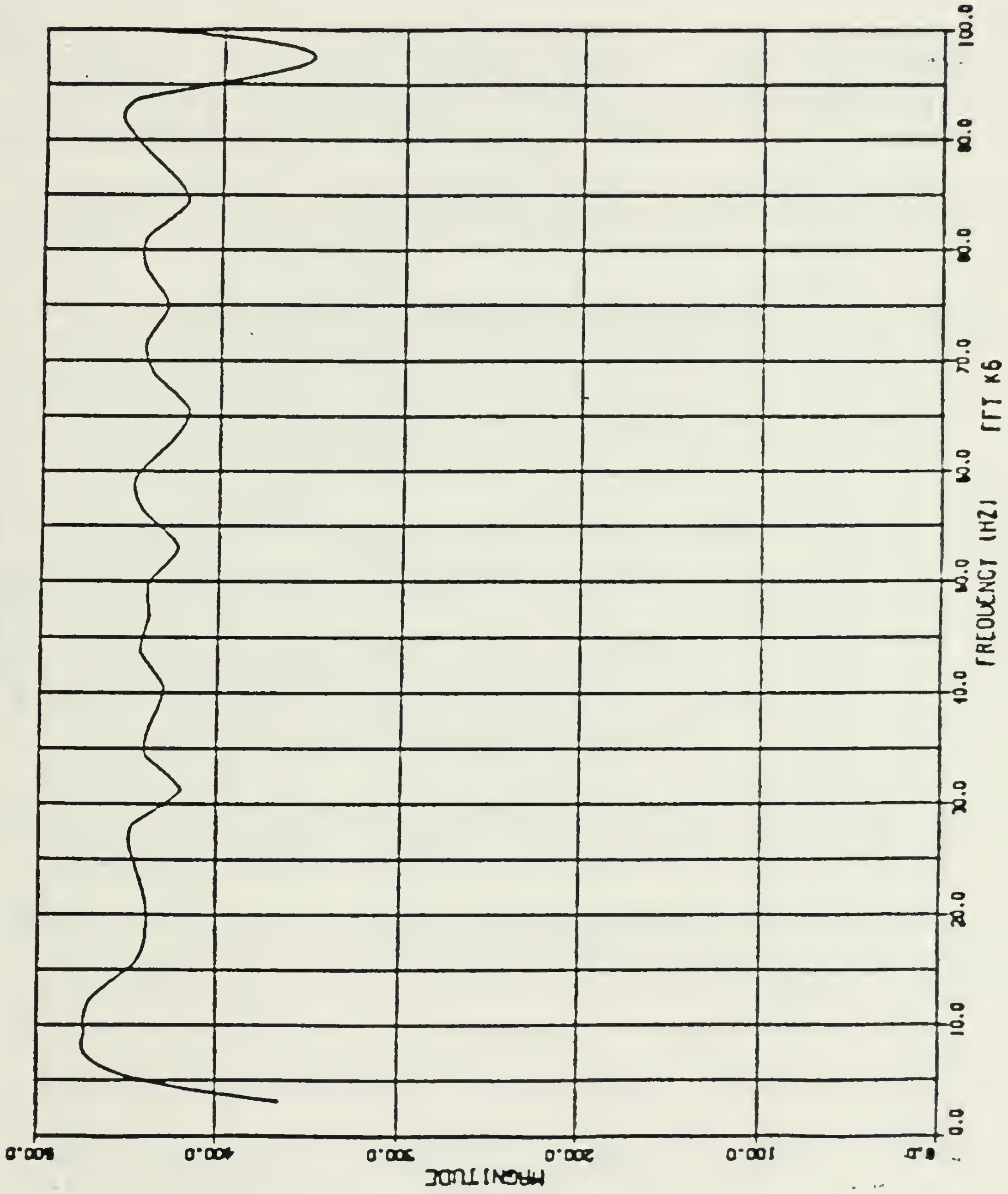


Figure O.5. Reflection Coefficient K6 for Utterence 'So What-MC'.

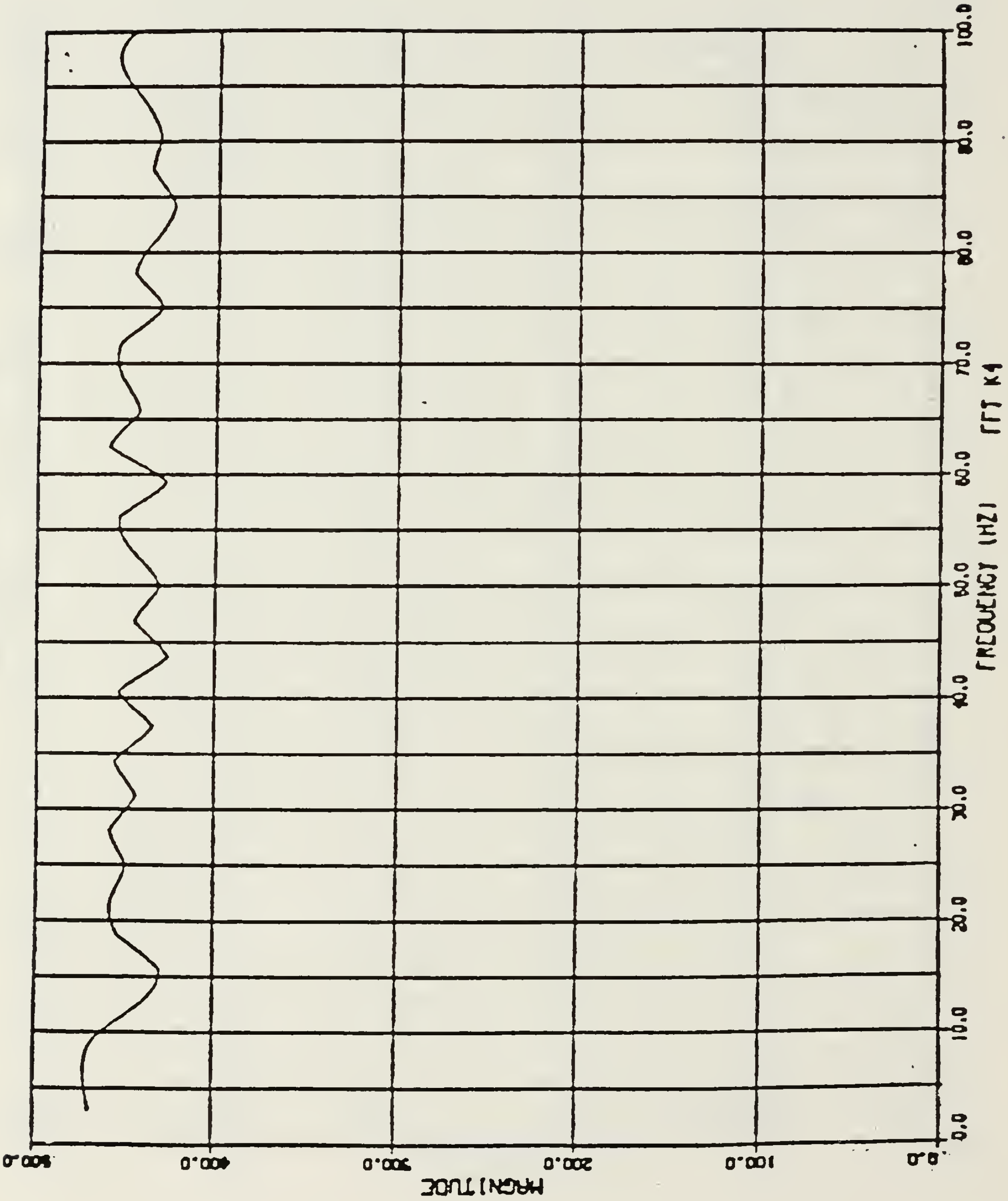


Figure O.6. Reflection Coefficient K4 for
Utterence 'So What-MC'.

LIST_OF_REFERENCES

1. "Report of the Panel on Communicative Disorders to the National Advisory Neurological and Communicative Disorders and Stroke Council,"Sensory_Aids_for_the Hearing_Impaired, IEEE Press, 1980.

2. J.M. Pickett, "Frequency Lowering for Hearing Aids," Sensory_Aids_for_the_Hearing_Impaired, IEEE Press, 1980.

3. G.T. Hall, Computer_Modeling_of_Voice_Signals_with Adjustable_Pitch_and_Formant_Frequencies, MSc Thesis, Naval Postgraduate School, Monterey, CA, 1976.

4. J.B. Evans, Speech_Synthesis_-_Models_and_Methods, IEEE 1983 Student Papers, 1983.

5. B. S. Atal and S. L. Hanauer, "Speech Analysis and Synthesis by Linear prediction of the Speech Wave," Speech_Analysis, IEEE Press, 1979.

6. J. Makhoul, "Linear Prediction: A Tutorial Review," Speech Analysis, IEEE Press, 1979.

7. G.H. Smith, Five_Voice_Synthesizers, Byte [illegible], Vol. 9, No. 10, September 1984.

8. B.S. Atal, Automatic_Speaker_Recognition_Based_on_Pitch Contours, PhD thesis, Polytech. Inst. Brooklyn, New York, 1968.

9. S.R. Parker, An_Intergrated_Approach_to_Discrete_[illegible] Processing_for_Scientists_and_Engineers, [illegible] notes, Naval Postgraduate School, Monterey, [illegible].

10. "Application Note: Number One", [illegible] Synthesis, [illegible].

11. [illegible] Flanagan, Speech_Analysis,_Synthesis_and Perception, Springer-Verlag, [illegible].

BIBLIOGRAPHY

Flanagan, J. L. and Rabiner, L. R., Speech_Synthesis, Dowden, Hutchingson & Ross, Inc., 1973.

Link_80_Operators_Guide, Digital Research, Monterey, CA, 1980.

Macro_Assembler_Operators_Guide, Digital Research, Monterey, CA, 1980.

Prolog_7804_Z80A_Processor_Counter/Timer_Card_User's_Manual, Prolog Corporation, Monterey CA, 1981.

Rabiner, L. R. and Shafer, R. W., Digital_Processing_of Speech_Signals, Prentice-Hall, 1978.

Shafer, R. W. and Markel, J. D., Speech_Analysis, IEEE Press, 1979.

Z-8400_Z80_CPU_Central_Processing_Unit,_Product_Specification, Zilog Inc., Santa Clara, CA, February 1981.

Zaks, R., Programming_the_Z80, Sybex Inc., 1982

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5100	2
2.	Department Chairman, Code 62 Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, California 93943	1
3.	Professor P. Moose, Code 62Me Naval Postgraduate School Monterey, California 93943	2
4.	Professor B. Madan Department of Computer Science and Engineering I.I.T., New Delhi - 110016 INDIA	1
5.	Lt. Vinny DiGirolamo, USN 3 Fire Road Drive Bay Shore, New York 11706	5
6.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	1

214173

Thesis
D5739 DiGirolamo
c.1 Analysis of a digital technique for frequency transportation of speech.